Universitat Politècnica de Catalunya
Programa de Doctorat de Matemàtica Aplicada

Departament de Matemàtica Aplicada III

Point-set Manifold Processing for Computational Mechanics:

thin shells, reduced order modeling, cell motility and molecular conformations

by Daniel Millán

Doctoral Thesis

Advisor: Marino Arroyo

Barcelona, July 2012

To Flavia

There is really no secret about our approach. We keep moving forward opening new doors and doing new things because we are curious. And curiosity keeps leading us down new paths. We are always exploring and experimenting. At WED[§], we call it Imagineering. The blending of creative imagination with technical know-how.

Walt E. Disney 1965 Presentation "Total Image"

[§]Disney called WED,"My back yard laboratory, my workshop away from work."

Abstract

Point-set Manifold Processing for Computational Mechanics: thin shells, reduced order modeling, cell motility and molecular conformations

Daniel Millán

In many applications, one would like to perform calculations on smooth manifolds of low-dimension d embedded in a high-dimensional space of dimension D. Often, a continuous description of such manifold is not known, and instead it is sampled by a set of scattered points in high dimensions. This poses a serious challenge. In this thesis, we approximate the point-set manifold as an overlapping set of smooth parametric descriptions, whose geometric structure is revealed by statistical learning methods, and then parametrized by meshfree methods. This approach avoids any global parameterization, and hence is applicable to manifolds of any genus and complex geometry. It combines four ingredients: (1) partitioning of the point set into subregions of trivial topology, (2) the automatic detection of the geometric structure of the manifold by nonlinear dimensionality reduction techniques, (3) the local parameterization of the manifold using smooth meshfree (here local maximum-entropy) approximants, and (4) patching together the local representations by means of a partition of unity.

We show the generality, flexibility, and accuracy of the method in four different problems. First, we exercise it in the context of Kirchhoff-Love thin shells, ($d = 2$, $D = 3$). We test our methodology against classical linear and non linear benchmarks in thin-shell analysis, and highlight its ability to handle point-set surfaces of complex topology and geometry. We then tackle problems of much higher dimensionality. We perform reduced order modeling in the context of finite deformation elastodynamics, considering a nonlinear reduced configuration space, in contrast with classical linear approaches based on Principal Component Analysis ($d = 2$, $D = 10,000$'s). We further quantitatively unveil the geometric structure of the motility strategy of a family of micro-organisms called Euglenids from experimental videos ($d = 1$, $D \approx 30,000$'s). Finally, in the context of enhanced sampling in molecular dynamics, we automatically construct collective variables, which characterize molecular conformations ($d = 1 \dots 6$, $D \approx 30$ to 1,000's).

Acknowledgments

I have really enjoyed doing imagineering under the supervision of Marino Arroyo, his guidance, freedom to work, criticism, wisdom of hope and optimism to try challenge problems make me more confident and patient to calm the anxiety for quick and shallow results. It has been a pleasure to be imbibed with his enthusiastic concept of research and work, an example I hope to follow during my career.

I am truly indebted and thankful to the faculty of LaCàN and Departament de Matemàtica Aplicada III, in particular, Professors Antonio Huerta and Antonio Rodríguez. Without their support, this project would not have been possible. I would also like to thank the reviewers of the thesis and the members of the committee for their useful comments and advice. I am very grateful to the people at LaCàN for providing such an enjoyable and stimulating working environment, my special thanks to David Modesto, Amir Abdollahi, Mohammad Rahimi, Christian Peco, Behrooz Hashemian, Kuan Zhang, David Prat, Susanta Ghosh, and last but not least to Juan Vanegas. A very special thanks to Adrian Rosolen, it has been a great experience to work with him and share his knowledge in many interesting technical discussions, and also his sincere friendship. Also, I would like to thank Javier Principe for his criticism and rigorous point of view, finally Anna Fabregas deserves all my grateful for beating the extremely annoying bureaucracy in a fast and kindly way.

During these years I have shared great time around Barna and Martorell with 'very important people', my outdoor friends, Lindaura, Matías, Herbert and Liliana, Giorgio (GG) and Hélène, Pachi and Ale, and Manel and Mónica. *Gracias totales*!... I would also comment that during these years I have really appreciated to live the most amazing history of the FCB, very thanks Pep and Lio. Visça el Barça!

I have so much to be grateful for my dear loved ones Flavi, Uli and Mica, who have been a source of unconditional inspiration and encouragement during *questa bella vita... è una catena ormai che scioglie il sangue dint'e vene sai.*

I want to thank the support of the European Commission (No. MIRGCT-2005-029158 and No. MIRG-CT-2005-029178), the Ministerio de Ciencia e Innovación (DPI2007-61054), and the European Research Council (240487).

Contents

Abstract v

Acknowledgments vii

Contents ix

List of Figures xi

Thesis body 1

1 Introduction . . . 3
2 State of the art in dimensionality reduction . . . 7
2.1 Linear dimensionality reduction . . . 7
2.2 Nonlinear dimensionality reduction . . . 8
2.3 Intrinsic dimensionality . . . 10
3 Manifold description from scattered points . . . 13
3.1 General setup . . . 14
3.2 Partition Of Unity . . . 16
3.3 Patch manifold learning and local parametrization . . . 17
3.4 Partition of unity to evaluate integrals on $\mathcal{M}$ . . . 21
3.5 Single representation of a manifold described by multiple patches . . . 21
4 Meshfree thin-shell analysis . . . 25
4.1 Introduction . . . 25
4.2 Kinematics of the shell . . . 26
4.3 Ritz-Galerkin discretization . . . 29
4.4 Numerical examples . . . 29
5 Model reduction . . . 37
5.1 Lagrangian mechanics in generalized coordinates . . . 38
5.2 Example: Neo-Hookean hyperelastic solid . . . 40
6 Quantitative analysis of the Euglenoid movement . . . 51
6.1 Quantitative observations . . . 51
6.2 Quantitative analysis of the shape space . . . 54
7 Describing molecular conformations with data-driven nonlinear collective variables . 59
7.1 Systematic determination of CVs . . . 59
7.2 Examples . . . 61

A Paper # 1: "Thin shell analysis from scattered points with *maximum-entropy* approximants" 77

B Paper # 2: "Nonlinear manifold learning for meshfree finite deformation thin shell analysis" **109**

C Paper # 3: "Nonlinear manifold learning for model reduction in finite elastodynamics" **145**

D Paper # 4: "Reverse engineering the euglenoid movement" **165**

E Auxiliary methods **191**
1 Affine Isometric Correction 193
2 Augmented Lagrangian for thin shells under Kirchhoff-Love assumptions 195

List of Figures

1 Schematic description of principal component analysis as the best linear fit. 7
2 Dimensionality reduction of an unstructured set of points sampling of a swiss roll in 3D. 10
3 Estimation of the intrinsic dimension for an unstructured set of points sampling a swiss roll in 3D. 11
4 Point-set surfaces that require partitioning either for their non-trivial topology or for its complex geometry. 13
5 Nonlinear dimensionality reduction of a truncated sphere. 14
6 Nonlinear dimensionality reduction from the quarter of a truncated sphere. 15
7 Illustration of the proposed method to compute the coarse partition of unity tied to the patches for a curve ($d = 1$) in the plane $D = 2$. 17
8 Visualization of a coarse partition of unity overlap regions on a sphere. 17
9 Embedding of a patch from a point-set surface. 18
10 Sketch of the method to smoothly parametrize point-set manifolds. 19
11 Local *max-ent* basis functions and spatial derivatives. 21
12 Schematic description of the different configurations of the shell. 27
13 Sketch of the pinched closed hemisphere shell test and convergence curves. 30
14 Reference configuration and deformed resulting surface for pinched close hemisphere. 31
15 Point-set for the example of the connected pipes. 31
16 Selected snapshots of the deformation process of the shell with complex topology. Non magnified deformation and colormap of the strain energy density. 32
17 Sketch for a brittle thin shell problem consisting of six connected open pipes. 33
18 Selected snapshots of the deformation process of a brittle thin shell with complex topology. 34
19 Sketch for the proof of concept neo-Hookean hyperelastic solid example in 2D. . . . 41
20 Estimation of the intrinsic dimension for the elastodynamics from a neo-Hookean hyperelastic body. 42
21 Two- and three-dimensional embeddings obtained with PCA, Isomap and NLIE. . . 43
22 Two-dimensional embedding Q obtained from the original training set X with NLIE, and subsampled node set $\hat{Q}$. 44
23 Reduced potential energy, total reduced mass, and reduced trajectory in the two-dimensional embedding from NLIE. 45
24 Time evolution of a high-dimensional trajectory $x(t)$, and the reduced trajectories $\mathbb{X}(q(t))$ obtained from PCA and NLIE. 46
25 Energy evolution for a test trajectory of the full system, and for the reduced trajectory. 47
26 Total energy evolution of the full system and total energy projected onto the reduced models by PCA for different numbers of eigenvectors. 47
27 Scanning electron micrographs of the pellicle surface of two flexible euglenids. 51
28 The time-reversibility of the Stokes equations in shape space. 52

29 Method for the quantitative analysis of video recordings of Euglenids undergoing metaboly. 53
30 Embedding of the frames in 3D for Euglenid #3 and parametrization as function of the number of bins. 55
31 Quantitative analysis of four video recordings of Euglenids undergoing metaboly. . . 56
32 Sketch of the slow manifold parametrization with nonlinear smooth collective variables. 60
33 Alanine dipeptide molecule, order parameter space and 3D embedding from Isomap. 62
34 After pruning of few conformations we compute the low-dimensional embedding from NLIE in 2D and the order parameter space. 63
35 Control points, to perform a fitting, in the embedded space for the alanine dipeptide aligned by using the three middle back bone atoms. Also, the mapping of low dimensional samples revel the slow manifold. 64
36 Control points, to perform a fitting, in the embedded space for the alanine dipeptide aligned by using all back bones. Also, the mapping of low dimensional samples revel the slow manifold. 65
37 Three-dimensional embedding and smooth parametrization for the titin I27 domain. 66

Thesis body

1 Introduction

In this dissertation, we address apparently unrelated topics: dimensionality reduction, meshfree analysis of thin shells, nonlinear model reduction of mechanical systems, quantitative analysis of a motility mode exhibited by the Euglenids, a family of protists, and the automatic detection of collective variables from biomolecular ensembles. These topics, as will become clear in the following sections, rely on the processing of manifolds represented by a set of scattered points in high dimensions.

Our brains understand the processes in Nature by organizing and interpreting stimuli. The perception of our environment is a complex process, by which we construct a mental representation that allows us to understand its behavior and how it evolves. For example, we are able to watch a complex turbulent flow and identify coherent structures, qualitatively predict the time evolution of these structures, and act accordingly. Our brain makes a very complex transformation of the massive amount of information from the sensory inputs, which can be thought of as points in high-dimension, and compresses it into a few perceptually meaningful features lying on, or close to, a low-dimensional manifold–the slow manifold. Comparing and classifying such observations, i.e. rationalizing our perceptions, depends crucially on modeling the nonlinear geometry of such manifold, embedded in the high-dimensional perception space. Many scientific endeavors generate enormous amounts of data, from numerical calculations in supercomputing platforms to high-throughput experimental measurements, facing researchers with a serious perception issue.

Over the last decades, a myriad of techniques in the area of statistical learning have emerged to extract the hidden variables from high-dimensional data sets, commonly as post-processing or data-exploration tools. While this has proven extremely useful in applications such as automatic perception (Tenenbaum et al., 2000; Roweis and Saul, 2000), climate science (Gámez et al., 2004), the study of the conformation dynamics of molecules (Das et al., 2006; Brown et al., 2008), or galaxy spectra classification (Vanderplas and Connolly, 2009), one would like to further process the slow manifold, for instance by finding low-dimensional representations of new out-of-sample data points or finding tangent vectors to it characterizing changes of the system.

Here, we propose a general methodology to perform calculations on smooth manifolds of low-dimension d embedded in a high-dimensional space of dimension D, and given by scattered points. We approximate the point-set manifold as an atlas of overlapping smooth parametric descriptions, whose geometric structure is revealed by statistical learning methods, and then parametrized by meshfree methods. This method closely mimics the theory of smooth manifolds. This approach avoids any global parametrization; it is applicable to manifolds of any dimension, any genus and complex geometry. We proceed in four steps:

1. Automatic partitioning of the point-set data into sub-regions of trivial topology by recursively application of graph based methods (Karypis and Kumar, 1998).

2. Automatic detection of the geometric structure of the manifold patches by either linear or nonlinear dimensionality reduction methods. These methods embed the high-dimensional training set in low dimensions, maintaining some of the geometric structure of the underlying manifold.

3. A smooth local parametrization is defined in the low-dimensional embedding of each patch. This can be realized with a variety of methods, from meshfree methods such as Moving Least Squares approximants (Lancaster and Salkauskas, 1981) to mesh-based methods such as subdivision finite elements in the case of surfaces (Cirak et al., 2000). Here the local *max-ent* approximants (Arroyo and Ortiz, 2006) are chosen, due to their smoothness, robustness, and applicability in any space dimension.

4. The local parametrizations are then glued together, if needed, with a Partition of Unity (PU) defined in the ambient space, which consequently is also a PU on the embedded manifold.

There have been extensions beyond linear representations of reduced models, *à la* Principal Components Analysis, in computational mechanics. For instance, Proper Generalized Decomposition methods (Chinesta et al., 2010) address high-dimensional problems by finding iteratively new basis functions of separated variables, resulting in a nonlinear approximation method. Farhat and co-workers have proposed methods to interpolate parametrized reduced models, which form a nonlinear manifold (Amsallem and Farhat, 2011).

The present thesis is organized around new developments in various fields, with manifold processing from scattered points as the unifying concept. In all the applications, numerical calculations on a smooth manifold of low-dimension embedded in high-dimensions are required. More specifically:

Thin-shell analysis. We apply the methodology to the geometrically exact theory of Kirchhoff-Love thin shells. In thin shells, the intrinsic dimension is known a priori, $d = 2$, and the shell deforms in three-dimensions, $D = 3$. Unlike previous meshfree methods, limited to very simple surfaces admitting a single parametrical space, the proposed method is very robust and general. As we will shown, it can deal very easily with shells of very complex geometry and topology. See Section 4 and Appendices A and B.

Model reduction. By taking advantage of the slow manifold parametrization, we develop the nonlinearly reduced dynamics of mechanical systems in a variational framework. We exemplify the method in finite deformation elastodynamics. In these problems, $D \approx 10^3 - 10^6$ while the slow-manifold on which the dynamics take place is expected to be rather low-dimensional and nonlinear. In our proof of concept test $D = 10^4$ and $d = 2, 3$. See Section 5 and Appendix C.

Motility of Euglenids. Swimming is generally accomplished by the repetitive execution of a path ($d = 1$) in shape space. By Purcell's scallop theorem (Purcell, 1977), at low Reynolds number such path needs to be non-reciprocal. We study the particular motility strategy of Euglenids, a family of unicellular eukaryotes, consisting of large amplitude highly concerted deformations of the entire body (euglenoid movement or metaboly). Unlike ciliary or flagellar motility, this mode is not well understood. We examine quantitatively video recordings of four Euglenids executing such motions with nonlinear dimensionality reduction methods. The low-dimensional embedding allows us to parametrize the geometric stroke smoothly in time and space, filtering the noise of the data, while retaining a sharp description of the stroke, a one-dimensional curve in $D \approx 10^4$, and allows us to analyze the hydrodynamics at low Reynolds number of this swimming strategy, its efficiency, discuss its role in the evolutionary history of this group of protists, and get inspiration of new technologies based on soft active surfaces. See Section 6 and Appendix D.

Molecular collective variables. We automatically identify and smoothly describe collective variables that best explain the conformational flexibility of molecules, such as proteins. We exercise the proposed methodology in *in silico* conformations from molecular simulations, although the method is general enough to be applied also in the study of *in vitro* conformations from nuclear magnetic resonance spectroscopy. The D-dimensional space describing these proteins is commonly around 10^3, whereas we find that the collective variables have dimensionality ranging from 1 to 6. See Section 7.

The structure of the thesis is as follows. The state of the art in methods for dimensionality reduction is presented in Section 2. Section 3 describes the core methodology to handle non-uniformly sampled point-set manifolds. Sections 4, 5 , 6, and 7 expand on the four specific areas of application mentioned above. These sections describe concisely the goals of the contribution, provide representative results, summarize the conclusions, and refer to (accepted or submitted) papers attached as appendices. In each of these sections, the most significant contributions are highlighted, and new research ideas are suggested.

Apart from these contributions, we outline below other collaborative research projects closely related to the topics of the thesis.

1. Developments in *max-ent* approximants (Rosolen et al., 2010, 2012). The collaboration has focused on the implementation of local *max-ent* (LME) and second order *max-ent* (SME) approximants. For instance, we have coded numerical integration rules in 1D, 2D and 3D, dual nearest neighbors lists and nodal spacing estimation for scattered set of points, and coding integration of both LME and SME approximants to perform meshfree thin shell analysis. Illustrative examples are included in Section 4 and Appendix A and B, which show the ability of *max-ent* approximants, e.g., to deal with higher order partial differential equations.

2. The algorithms to compute the *max-ent* approximants have been implemented both in MatLab and as a part of SolMec, a C++ library (Rosolen et al., 2008).

3. We are currently collaborating with Professor Timon Rabczuk and Fatemeh Amiri at Bauhaus University Weimar in Germany. We are developing phase-field models of fracture in linear thin shells, see the ongoing work in Section 4. We have developed a new C++ library, ShellMec, for this purpose.

2 State of the art in dimensionality reduction

The field of Statistical Learning has emerged over the last decades to address the challenge of extracting meaningful information from empirical data arising in diverse areas of science and engineering. One of its main areas of application is *dimensionality reduction*, also known as manifold learning. The key purpose of this field is to circumvent the curse of dimensionality in high-dimensional data sets. This is done by finding a lower $d-$dimensional representation, $d \ll D$, which captures the most relevant features of the data. In other words, these methods identify the hidden variables, which best explain the behavior of a given system.

> *Dimensionality reduction is the procedure, which given a input data set* $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\} \subset \mathbb{R}^D$*, finds a lower dimensional representation of it,* $\Xi = \{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_N\} \subset \mathbb{R}^d$ *with* $d < D$*, according to some criterion* (Fodor, 2002).

2.1 Linear dimensionality reduction

The most extended technique to perform dimensionality reduction was introduced in 1901 by Pearson (1901). In this seminal work, Pearson developed a method to obtain the best linear fit, which minimizes the distance to the original data, see Figure 1. Throughout the years this method has been rediscovered and extended in many areas. It is known by different names, such as Principal Components or Hotelling Transform (Hotelling, 1933), Karhunen-Loève Transform (Karhunen, 1946; Loève, 1955), Empirical Orthogonal Functions (Lorenz, 1956) and Proper Orthogonal Decomposition (Lumley, 1967). Here we will refer to this method as Principal Component Analysis (PCA). A modern reference to this subject is Jolliffe (2002).

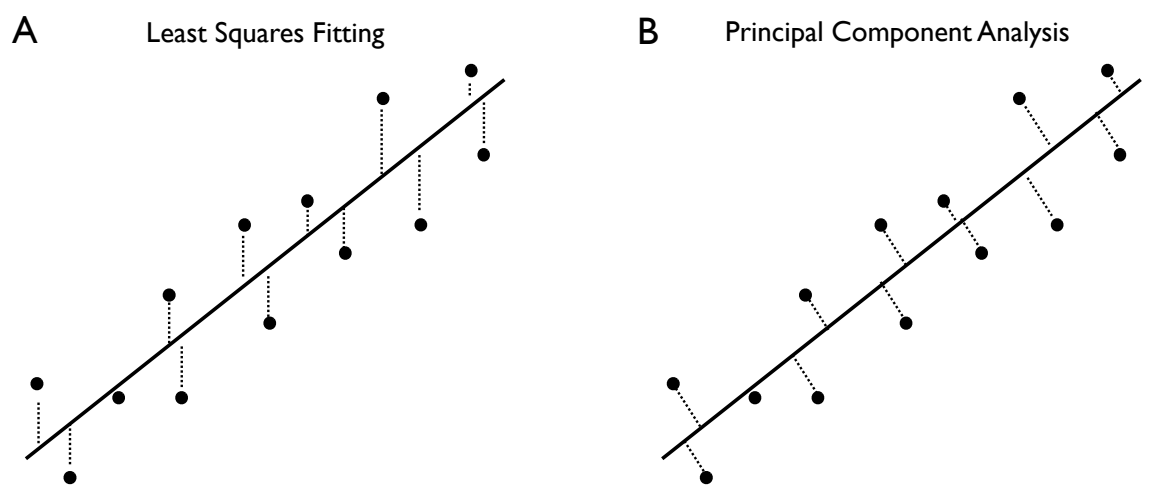


Figure 1: (A) Least squares and (B) principal component analysis fits to a set of scattered points in 2D. The least squares fit depends on the choice of axis and requires solving a linear system of equations, while the PCA is geometrically objective and requires finding eigenvalues and eigenvectors.

PCA is a standard tool in data analysis (Jain et al., 2000), computer graphics (Pauly, 2003), manifold learning (Zhang and Zha, 2005), or model reduction techniques in computational mechanics (Krysl et al., 2001; Niroomandi et al., 2010). PCA identifies the low $d-$dimensional affine subspace that best explains the variance of a higher $D-$dimensional data set. The original data is transformed to a new orthogonal coordinate system such that the projection of the data on the subspace defined by the first d coordinate directions, $d \leq D$, maximizes the variance as compared to any other

projection onto a d-dimensional subspace. Algebraically, given a data set in high dimension $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\} \subset \mathbb{R}^D$, we define the matrix $\boldsymbol{X} \in \mathbb{R}^{D\times N}$ formed by the centered point coordinates as column vectors, $\boldsymbol{x}_a - \bar{\boldsymbol{x}}$ for $a = 1, \ldots, N$. Here, $\bar{\boldsymbol{x}}$ is the average of the points

$$\bar{\boldsymbol{x}} = \frac{1}{N}\sum_{a=1}^{N} \boldsymbol{x}_a.$$

The covariance matrix is then

$$\boldsymbol{C} = \frac{1}{N}\boldsymbol{X}\boldsymbol{X}^T \quad \in \mathbb{R}^{D\times D}.$$

This positive (semi-)definite symmetric matrix has real eigenvalues and diagonalizes in an orthonormal basis of eigenvectors. We define $\boldsymbol{V} \in \mathbb{R}^{D\times d}$ as the eigenvector matrix formed by the d eigenvectors corresponding to the largest d eigenvalues. These eigenvectors form an orthogonal basis of the Euclidean structure of the reduced manifold. This hyper-plane passes through $\bar{\boldsymbol{x}}$ and is parallel to these first d eigenvectors. The matrix $\boldsymbol{V}$ defines an orthogonal projection Π_{PCA} relative to $\bar{\boldsymbol{x}}$ onto the reduced space of dimension d, i.e.

$$\begin{aligned} \Pi_{PCA}: \quad \mathbb{R}^D &\longrightarrow \mathbb{R}^d \\ \boldsymbol{x} &\longmapsto \boldsymbol{V}^T(\boldsymbol{x} - \bar{\boldsymbol{x}}). \end{aligned}$$

Another well-known method to perform linear dimensionality reduction is Multidimensional Scaling (MDS). This method was developed, and currently is a standard tool, in the fields of psychophysics and sensory analysis. MDS is a more suitable technique when, instead of data coordinates, one has similarity or disimilarity scores for each pair of *stimuli*. Here we refer to classical metric Multidimensional Scaling (Torgerson, 1952). Extensions and details to non-metric MDS can be found for instance in Cox and Cox (2001) and references therein.

The idea behind the method is to project the data set onto an affine space in such a way that the isometry is preserved as much as possible, as compared to the input distance matrix. Operationally, this requires finding eigenvalues and eigenvectors of an $N \times N$ matrix. If the Euclidean metric is used to compute the distance between pairs of realizations, then this method is equivalent to PCA. Thus, PCA and MDS share are intimately related; they both define a continuous linear implicit mapping, and the core operation is computing eigenvalues and eigenvectors of a full matrix. If data is not too high dimensional but the number of points N is large, PCA is better-suited since it involves a $D \times D$ matrix. By contrast, MDS is better when dimensionality is very high but the number of points rather low. PCA and MDS have successfully shown their capabilities to capture the Euclidean structure underlying data in many applications, but fail to produce compact and useful representations in others, presumably because of the presence of strong nonlinear correlations.

2.2 Nonlinear dimensionality reduction

Nonlinear Dimensionality Reduction methods (NLDR) have been developed over the last decades to address complex nonlinear problems in areas such as multivariate data analysis (Sammon, 1969), image processing (Hérault et al., 1997, 1999), and pattern recognition (Jain et al., 2000). For an extended NLRD literature and insightful remarks, the reader is referred to the book by Lee and Verleysen (2007) or the report by van der Maaten et al. (2009). Recently, two methods made a breakthrough in this field: isometric mapping (Isomap (Tenenbaum et al., 2000)) and Local Linear Embedding (LLE (Roweis and Saul, 2000, 2003)). Each of these methods has received about 4,000 citations in ten years. They have widen the range of application, and have motivated the development

of new NLDR methods, which exploit the intrinsic capabilities of Isomap and LLE to manage data lying on nonlinear manifolds (Zhang and Zha, 2005; de Silva and Tenenbaum, 2003; Choi and Choi, 2007; Donoho and Grimes, 2003; Belkin and Niyogi, 2003; Chang and Yeung, 2006; Zhang and Wang, 2007). These techniques have been applied successfully in climate data analysis (Gámez et al., 2004), the study of the conformation dynamics of molecules (Das et al., 2006; Brown et al., 2008), and galaxy spectra classification (Vanderplas and Connolly, 2009).

Isomap

Isomap is based on MDS, but uses the geodesic distance instead of the Euclidean one to build the distance matrix. The geodesic distance between pairs of points on the manifold is approximated as the shortest path distance in a graph of $k-$neighbors. Isomap is more robust than other NLDR methods for data polluted with noise, or for data-sets that are non-uniformly distributed. The very goal of Isomap, i.e. finding near-isometric low-dimensional embeddings, clashes with Gauss's Theorema Egregium for surfaces or more recent results on isometric low-dimensional embeddings of higher-dimensional manifolds. As a consequence, it can become excessively frustrated for highly curved manifolds and become unstable. Moreover, Isomap cannot deal with non-convex manifolds, and can be computationally very demanding when the number of points is large.

LLE

By the very definition of a manifold, it should locally look like Euclidean space. LLE locally approximates the manifold by a small patch, such that each element of the manifold $\boldsymbol{x}_a$, $a = 1, \ldots, N$ is reconstructed from its nearest neighbors. The low-dimensional LLE embedding tries to preserve these relations. LLE does not try to be isometric and, in fact, ignores metric information altogether by producing an embedding in low-dimension of unit covariance. Complete details are given in Roweis and Saul (2000, 2003). A short but mathematically concise description of the method can be found in the technical report by de Ridder and Duin (2002).

LLE is more sensitive to the number of neighbors than Isomap. However, from an efficiency viewpoint, in contrast with Isomap, the $N \times N$ matrix whose eigenvalues and eigenvectors need to be computed is sparse for the LLE method. Therefore, this method is applicable to problems with large numbers of sampling points. LLE presents instabilities due to the ill-conditioning of the optimization problem, leading often to spurious distortions. The stability of the LLE method has been enhanced in subsequent modifications, such as Hessian LLE (Donoho and Grimes, 2003), or the Modified LLE method (MLLE, Zhang and Wang (2007)). These methods produce better quality low-dimensional embeddings in general, and the MLLE method has a small computational overhead as compared to LLE.

For concreteness, we exemplify these techniques by considering the dimensionality reduction of an unstructured set of points sampling a swiss roll in 3D, see Fig. 2A. This is a classical benchmark test to assess the capabilities of NLDR techniques. Figure 2B shows the two-dimensional embedding with PCA, which cannot unfold the sampled manifold and collapses points into a line. On the other hand, the NLDR methods clearly reveal the underlying two-dimensional geometry, see Figs. 2C,D. Figure 2C shows how the LLE based techniques do not preserve the isometry; they introduce high distortions because they force the low-dimensional embedding to have unit covariance. Finally, Isomap retrieves a low-dimensional embedding that preserves the geometric structure and the local metric, which is possible for this surface of zero Gaussian curvature Fig. 2D.

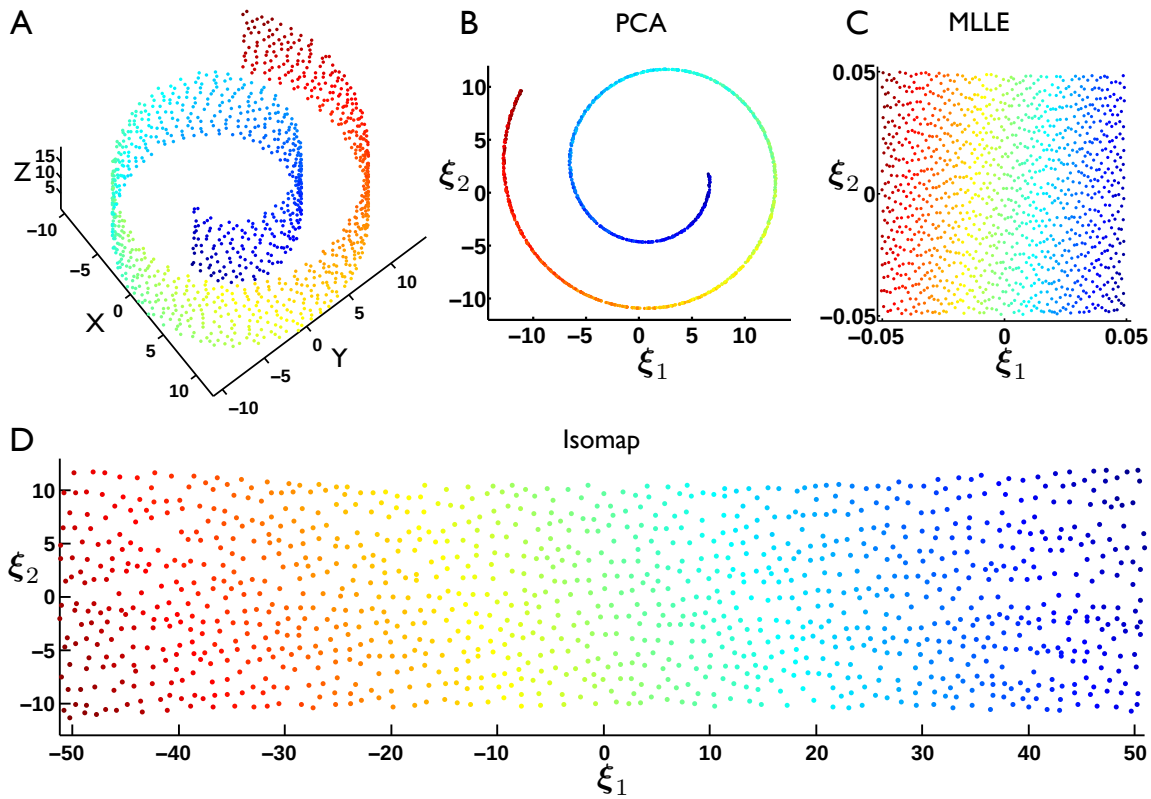


Figure 2: Classical benchmark in dimensionality reduction. (A) Sampling of a swiss roll in 3D by an unstructured set of points. Two-dimensional embeddings obtained with PCA (B), MLLE (C), and Isomap (D). Color map is a visual guide. The failure of PCA is apparent.

2.3 Intrinsic dimensionality

In the NLDR algorithms outlined above, the dimensionality of the manifold, d, is assumed to be given. However, in practical applications, the manifold dimensionality may not be known, and we cannot resort to visual inspection in high dimensions. The estimation of the intrinsic dimensionality underlying a high-dimensional data set is an important question in data exploration. Here we give a brief account of three methods to estimate d (see Lee and Verleysen (2007) for details and other methods):

- *The correlation dimension* method considers a closed ball of radius ε at the center of each data point, and counts the number of points inside of this ball. The dimension is estimated by noting that the average number of counts $C(\varepsilon)$ should grow linearly with ε for 1D objects, quadratically for 2D entities, and so on, i.e. $C(\varepsilon) \propto \varepsilon^d$. The correlation dimension is then defined as $d_{corr}(\varepsilon) = d \log C(\varepsilon)/d \log \varepsilon$. It is a scale-dependent quantity that should be interpreted carefully a: at very large ε, $d_{corr} \approx 0$ as the data set looks like a single point, while for small ε, the dimension is overestimated by the presence of noise or poor sampling. Details about practical implementations and insightful remarks can be found in the book by Lee and Verleysen (2007) and references therein.

- *Local PCA* analyzes small local subsets of the data. The intrinsic dimension d for each of these patches is selected such that it preserves a given fraction of the variance of the original data,

e.g.

$$0.95 \leq \frac{\sum_{i=1}^{d} \lambda_i}{\sum_{i=1}^{D} \lambda_i},$$

where λ_i are the eigenvalues of the covariance matrix arranged in decreasing order. In contrast with the global correlation dimension, this method provides a local estimation of the intrinsic dimension.

- *The reconstruction error* is a measure of the dissimilarity between the original high-dimensional data points $\boldsymbol{x}_a \in \mathbb{R}^D$ and their reconstruction from the embedded data, generally computed as a weighted average of the neighbors of $\boldsymbol{\xi}_a \in \mathbb{R}^d$. While for PCA the notion of reconstruction is straightforward, this is not the case for NLDR methods. For instance, we can then define the relative reconstruction error for PCA as follows

$$e = \frac{1}{N} \sum_{a=1}^{N} \frac{\left| \boldsymbol{x}_a - \boldsymbol{V}\boldsymbol{V}^T(\boldsymbol{x}_a - \bar{\boldsymbol{x}}) - \bar{\boldsymbol{x}} \right|}{|\boldsymbol{x}_a|}. \tag{1}$$

These three techniques are easy to implement, and also robust for data not polluted with noise. On the other hand, the notion of intrinsic dimension becomes strongly scale dependent for noisy measurements or poor sampled regions, and then these methods cannot be used as a black box. In such situations, physical understanding of the system should be combined with dimension estimation methods. For instance, we present in Fig. 3 the results of these methods for an unstructured set of points sampling a swiss roll (input data shown in Fig. 2A).

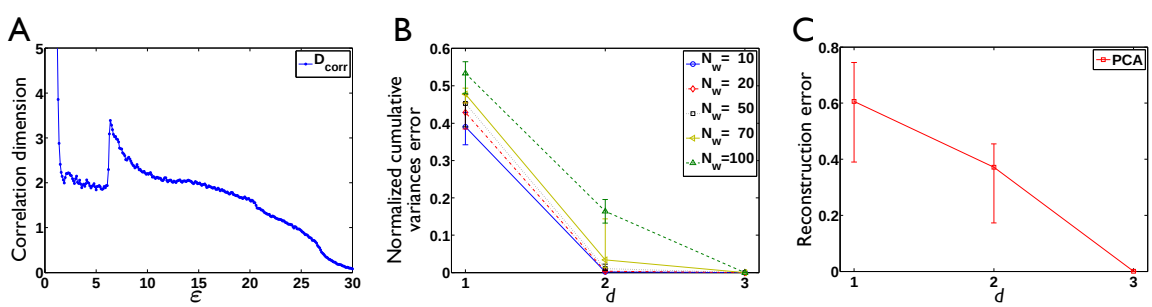


Figure 3: Estimation of the intrinsic dimension for an unstructured set of points sampling a swiss roll in 3D, see Fig. 2A. (A) Correlation dimension is a scale-dependent quantity that should be interpreted carefully. We can distinguish three different scales: the first one, $\varepsilon < 1.5$, gives us spurious high values of the intrinsic dimension, because of the poor sampling at low scales; the second one computes the right dimension of the embedded low-dimensional point-set manifold ($1.5 < \varepsilon < 6$) ; and finally, for $\varepsilon > 6$, the correlation dimension abruptly raises due to counts coming from the other sheets in the folded surface, and then decreases monotonically down to zero because at large scales, the manifold looks like a point. (B) Mean of the normalized cumulative variance error from local PCA ($\pm$ 25th and 75th percentiles), $1 - \left(\sum_{i=1}^{d} \lambda_i / \sum_{i=1}^{D} \lambda_i\right)$, computed in small local subsets of the data, where N_w is the window size, and λ_i denote the eigenvalues of the local covariance matrix arranged in decreasing order. The intrinsic dimension d is selected such that it preserves a given fraction of the variance of the original data, and is correctly estimated to be 2. In contrast with the global correlation dimension, this method provides a local estimation of the intrinsic dimension. (C) Mean of the reconstruction error as defined in Eq. (1) ($\pm$ 25th and 75th percentiles). The reconstruction error is a measure of the dissimilarity between the original high-dimensional data points and their reconstruction from the low-dimensional embedding. As we expect PCA is not able to detect the right intrinsic dimension.

3 Manifold description from scattered points

In this section, we introduce a novel technique to perform numerical calculations on smooth manifolds described in terms of a set of scattered points. We first illustrate in a toy example some challenges that need to be addressed to use a low-dimensional embedding from a NLDR technique as a parametric patch, see Fig. 4. Later on, the main ingredients of the method are described.

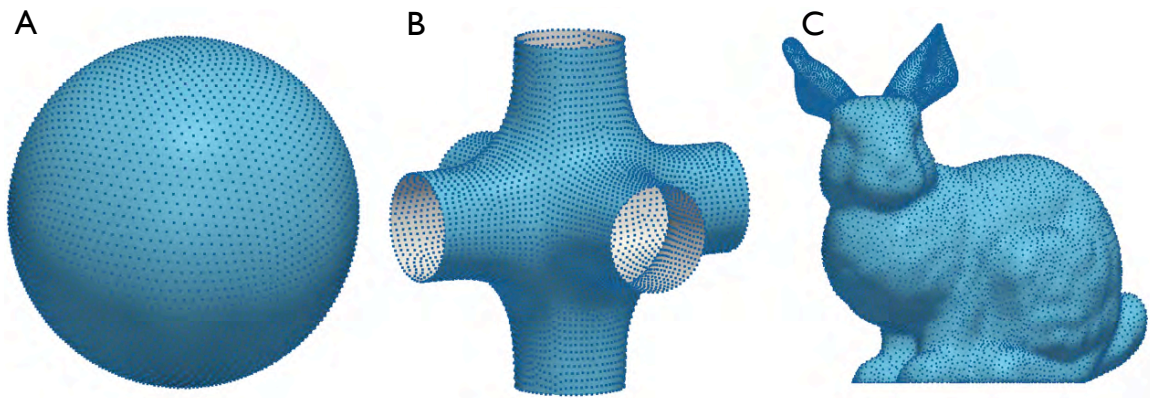


Figure 4: Three point-set surfaces that require partitioning for different reasons: (A) and (B) for their non-trivial topology, and (C) for its complex geometry. As noted in Levin (2003), a fundamental difficulty in defining basis functions and performing calculations on an embedded surface, as compared to open subsets in Euclidean space, is the absence in general of a single parametric domain. A simple example is the sphere, which does not admit a single singularity-free parametrization. Mesh-based methods, consisting of a collection of local parametrizations from the parent element to the physical elements, do not have any difficulty in this respect at the expense of reduced smoothness across the element boundaries or the need for special techniques. In meshfree methods, such a natural parametric domain is not available, and the description of surfaces with a topology different to that of an open set in $\mathbb{R}^2$, such as a sphere (A) or a set of connected pipes (B), is a challenge. Even for surfaces homeomorphic to open two-dimensional sets, such as that depicted in (C), the geometric complexity can make it very difficult to produce well-behaved global parametrizations.

Partition of the data set

As mentioned in Section 2, the hidden low-dimensional embedding of a nonlinear manifold can be learned with NLDR techniques. Unfortunately, these methods may experience some difficulties, even in apparently simple problems. Let us to consider for instance a set of uniformly distributed points on a large portion of a sphere, as illustrated in Fig. 5A. The points clearly represent a surface with the topology of a two-dimensional open set. However, Isomap is not able to embed this cloud of points in two-dimensions. Isomap fails due to the high frustration arising from the impossibility to embed isometrically such a large portion of a sphere. On the other hand, LLE provides satisfactory results, at the expense of generating very large local distortions, see Fig. 5B, which are unavoidable and a corollary of Gauss' Theorema Egregium. Another notable feature of the results of LLE in this example is the lack of a clear metric relationship between the two-dimensional embedding and the original data, as noted earlier. Both the distortions and the lack of metric correspondence are problematic for our purposes. Indeed, a good quality sampling of a surface may become strongly distorted or scaled in one direction if the surface is elongated. As a result, the H^1 norm of the parametrization from the two-dimensional embedding into three-dimensions may become very large and non-uniform, leading to inaccurate numerical calculations.

We address the unavoidable distortions by partitioning the original data set into a small number of patches, see Fig. 6A. We then embed each partition in two dimensions with NLDR techniques. We use the METIS library to partition the data (Karypis and Kumar, 1998). Partitioning is not only advisable to obtain good quality embeddings of low-dimension; it becomes unavoidable for point-set manifolds of general topology, as NLDR methods are applicable only to manifolds of trivial topology. It is always possible to recursively partition a data set until all the partitions have trivial topology. Figures 6B,C,D show the low-dimensional embedding generated by Isomap, LLE and MLLE. Isomap provides a good quality embedding of low-dimension, while LLE produces spurious distortions. MLLE corrects these distortions, yet introduces a uniform stretching of the domain in one direction as a result of its affine invariance and the unit covariance constraint.

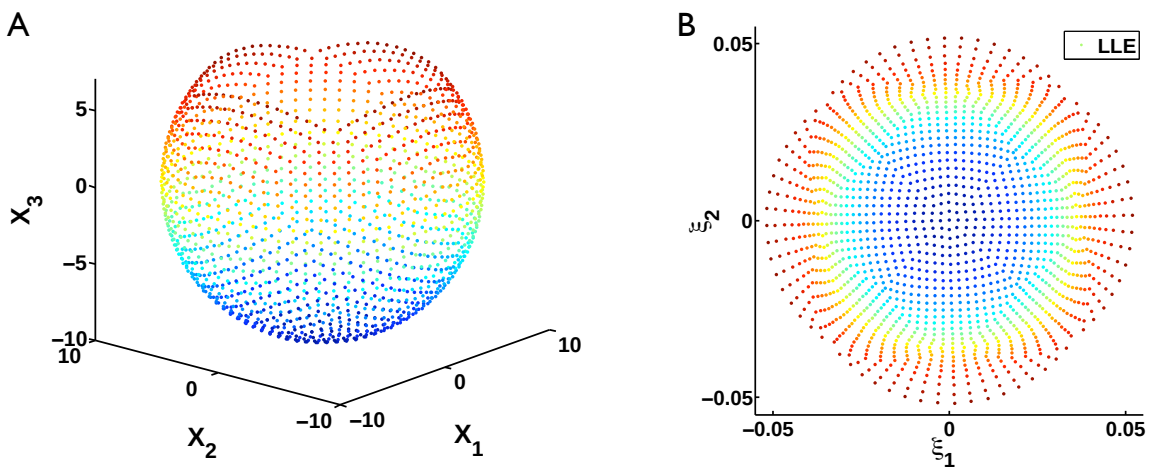


Figure 5: (A) Set of scattered points quasi-uniformly distributed on a truncated sphere. (B) Two-dimensional embedding obtained by LLE, exhibiting unavoidable high distortions. The color coding is a visual guide.

To deal with the uniform distortions of LLE-based methods, i.e. the lack of metric correspondence between the low-dimensional embedding and the original data, we propose a simple iterative method that we call Affine Isometric Correction (AIC). This method finds the affine transformation that minimizes a stress function, which measures the isometry. Full details can be found in Appendix E1. The low-dimensional embeddings resulting from post-processing the raw NLDR results with AIC are shown in Figs. 6E,F,G, and the values of the stress function are given in Table 1. It is clear that this simple procedure significantly improves the isometric quality of LLE-based methods a posteriori, while it only changes marginally the results of metric methods such as Isomap. In subsequent calculations, we choose the MLLE method combined with AIC for treating point-set surfaces not affected with noise, as it provides good quality embeddings of low-dimension, metrically related to the original data, and it involves calculations on sparse matrices. As will be clear later, in applications in much higher dimension, possible affected by noise, we use Isomap to compute the seed of an in-house iterative NLDR method that we call Nonlinear Locally Isometric Embedding (NLIE), see Appendix C.

3.1 General setup

Let $P = \{\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_N\} \subset \mathbb{R}^D$ be a set of control points sampling a smooth $d-$manifold $\mathcal{M}$ embedded in $\mathbb{R}^D$, and possibly affected by noise (the training point set). The goal is to obtain

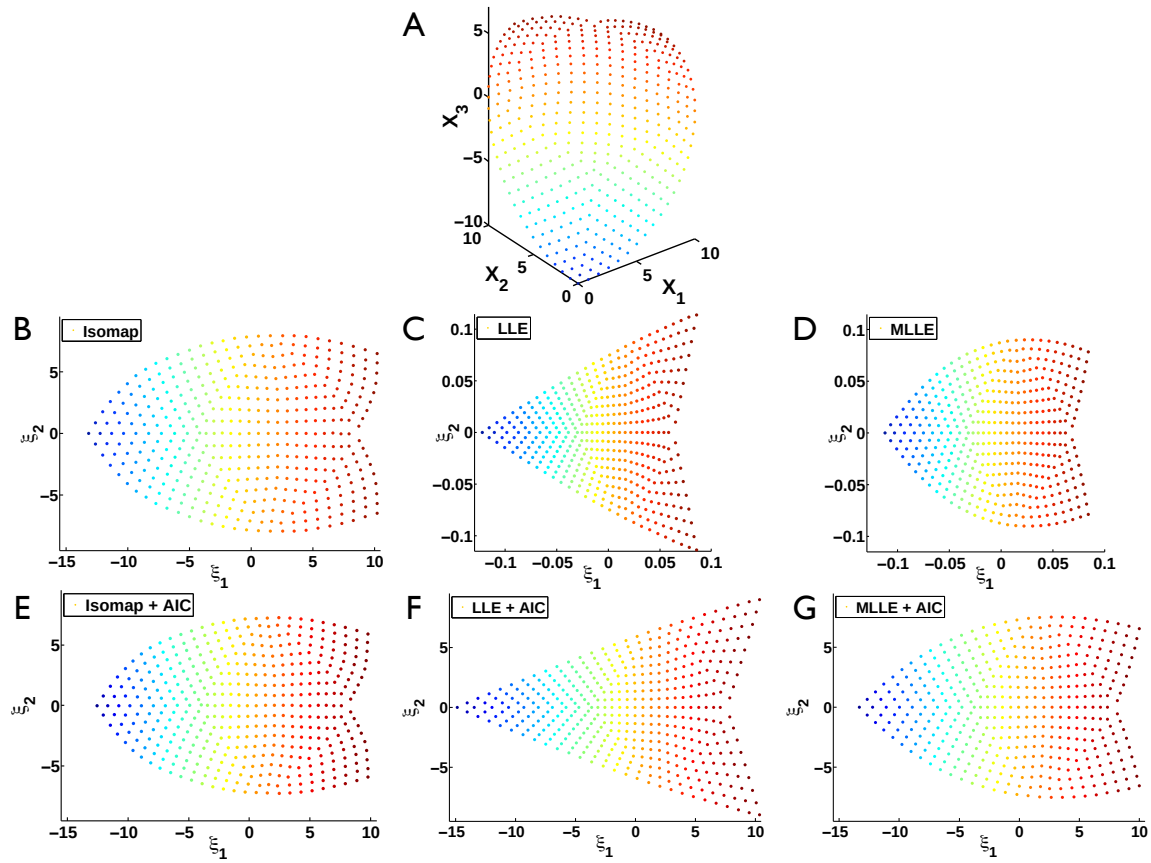


Figure 6: (A) Partition of the data set in Fig. 5. (B-C-D) Two-dimensional embeddings given by a direct application of Isomap, LLE and MLLE. (E-F-G) Two-dimensional embeddings given by Isomap, LLE and MLLE after the affine isometric correction step.

Table 1: Stress energy cost function for the quarter of a truncated sphere, see Figure 6 and Appendix E1.

	PCA	Isomap	LLE	MLLE
Original	69.1	25.9	30.6×10^4	26.8×10^4
AIC	45.1	11.7	96.5	6.22

a numerical representation of $\mathcal{M}$, which we denote by $\mathcal{M}^h$. We obviously require that the node separation and the noise be small enough for the manifold to be recognizable.

We consider another set of geometric markers, $Q = \{\boldsymbol{Q}_1, \boldsymbol{Q}_2, \ldots, \boldsymbol{Q}_M\} \subset \mathbb{R}^D$, typically a subset of P but not necessarily. We partition these geometric markers into L groups on the basis of proximity (METIS domain decomposition with a k-nearest neighbor graph). We represent these groups of geometric markers with index sets $\mathcal{I}_\kappa, \kappa = 1, \ldots, L$ with $\cup_{\kappa=1}^{L} \mathcal{I}_\kappa = \{1, 2, \ldots, M\}$ and $\mathcal{I}_\kappa \cap \mathcal{I}_\iota = \emptyset$ when $\kappa \neq \iota$, and use Greek subindices to refer to entities associated with these groups of markers. As it will become clear below, there is a one-to-one correspondence between these groups of geometric markers and the local parametrizations of the surface, which we call patches. For simplicity, we will denote the points in P and its associated objects with a lower case subindex, e.g. $\boldsymbol{P}_a$, for $a = 1, 2, \ldots, N$,

and the geometric markers in Q and its associated objects with an upper case subindex, e.g. $\boldsymbol{Q}_A$, for $A = 1, 2, \ldots, M$ $(M \leq N)$.

3.2 Partition Of Unity

In order to patch together local representations of the manifold, we introduce the concept of partition of unity, which is a fundamental tool in the theory of differential geometry of manifolds (do Carmo, 1976), see Box 1. A differentiable manifold $\mathcal{M}$ is an abstract mathematical space that locally resembles Euclidean space, and has a global differential structure enabling calculus on it.

Suppose that $\mathcal{M}$ is a differentiable manifold of class C^n, where $0 \leq n < \infty$. Let $\{U_\alpha\}$ be an open covering of $\mathcal{M}$. Then a partition of unity subordinate to the cover $\{U_\alpha\}$ is a collection of nonnegative real-valued C^n functions ψ_κ on $\mathcal{M}$ satisfying the conditions

- The supports of the ψ_κ are compact and locally finite
- The support of ψ_κ is completely contained in U_α for some α
- The ψ_κ sum to one at each point of $\boldsymbol{x} \in \mathcal{M}$: $\sum_\kappa \psi_\kappa(\boldsymbol{x}) = 1$

Note that this last condition is actually a finite sum at each point because of the local finiteness of the supports of the ψ_κ. We are particularly interested in the case where the number of coverings and the partition of unity functions is the same.

Box 1: Partition of unity on a differentiable manifold $\mathcal{M}$

Shepard partition of unity

We consider a Shepard partition of unity associated with the geometric markers. Consider a set of non-negative reals $\{\beta_A\}_{A=1,2,\ldots,M}$ associated with each point in Q. We define the Shepard partition of unity with Gaussian weight associated to the set Q as the functions $w_A : \mathbb{R}^D \to \mathbb{R}$ given by

$$w_A(\boldsymbol{x}) = \frac{\exp(-\beta_A \left|\boldsymbol{x} - \boldsymbol{Q}_A\right|^2)}{\sum_{B=1}^{M} \exp(-\beta_B \left|\boldsymbol{x} - \boldsymbol{Q}_B\right|^2)}.$$

For efficiency, and given the fast decay of the Gaussian functions, these functions are numerically treated as compactly supported.

We aggregate these partition of unity functions by patches, as depicted in Fig. 7 for $d = 1$ and $D = 2$, yielding a coarser partition of unity given by the functions

$$\psi_\kappa(\boldsymbol{x}) = \sum_{A \in \mathcal{J}_\kappa} w_A(\boldsymbol{x}).$$

These functions form a partition of unity in $\mathbb{R}^D$, and consequently also in $\mathcal{M}$. We consider the index sets of all control points influencing each patch, $\mathcal{J}_\kappa$, with $\cup_{\kappa=1}^{L} \mathcal{J}_\kappa = \{1, 2, \ldots, N\}$, but now $\mathcal{J}_\kappa \cap \mathcal{J}_\iota \neq \emptyset$ due to the overlap between patch partition of unity functions. Roughly speaking, these sets are $\{a \mid \boldsymbol{P}_a \in \operatorname{sup} \psi_\kappa\}$, slightly enlarged so that the patch parametrization is smooth on the boundary of the support of ψ_κ. Note that, as can be seen in Fig. 7, the partition unity functions ψ_κ restricted to the manifold are quite insensitive to the position of the geometrical markers $\boldsymbol{Q}_A$ in the direction normal to the curve. Figure 8 illustrates the overlap regions for a partition of the sphere.

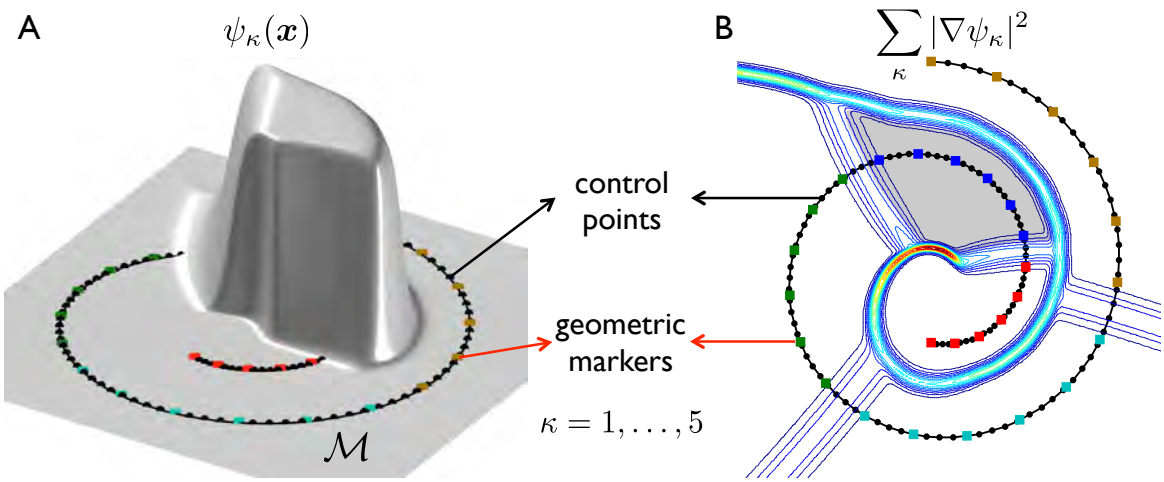


Figure 7: Illustration of the proposed method for a curve ($d = 1$) in the plane $D = 2$. (A) Illustration of a function $\psi_\kappa(\boldsymbol{x})$ of the coarse partition of unity tied to the patches. (B) Visualization of the coarse partition of unity overlap regions. The partition of the geometric markers is color-coded.

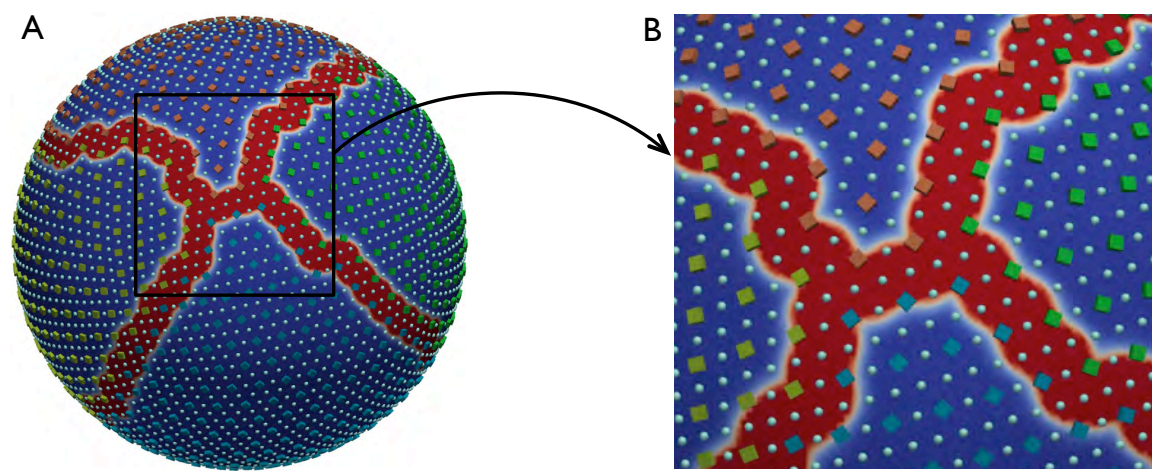


Figure 8: (A) Visualization of a coarse partition of unity overlap regions on a sphere (contour map of $\sum_\kappa |\nabla \psi_\kappa|^2$). (B) Zoom from a selected region on the sphere, it can be appreciated the partitioning of the geometric markers ($\diamond$, color-coded by the partitions), the width of the overlapping, and the control points $\circ$.

3.3 Patch manifold learning and local parametrization

For each patch, we obtain a low-dimensional embedding R_k of the points $P_\kappa = \{\boldsymbol{P}_a\}_{a \in \mathcal{J}_\kappa} \subset \mathbb{R}^D$ with a NLDR technique, resulting in the representation $\Xi_\kappa = \{\boldsymbol{\xi}_a\}_{a \in \mathcal{J}_\kappa} \subset \mathbb{R}^d$, this is

$$\begin{aligned} R_\kappa : \mathbb{R}^D &\longrightarrow \mathbb{R}^d \\ P_\kappa &\longmapsto \Xi_\kappa. \end{aligned}$$

For instance, in meshfree thin shell analysis MLLE with AIC, produces in general a low-dimensional embedding of good geometric quality and can be computed efficiently. See Fig. 9 for an illustration in a complex example. If the automatic partitioning of the data creates patches of complex topology,

e.g. a tubular partition in an ear of the bunny in the figure, or patches of high geometric complexity leading to excessive geometric distortions, such as the blue partition at the tip of one ear, we proceed by recursively re-partitioning such patches.

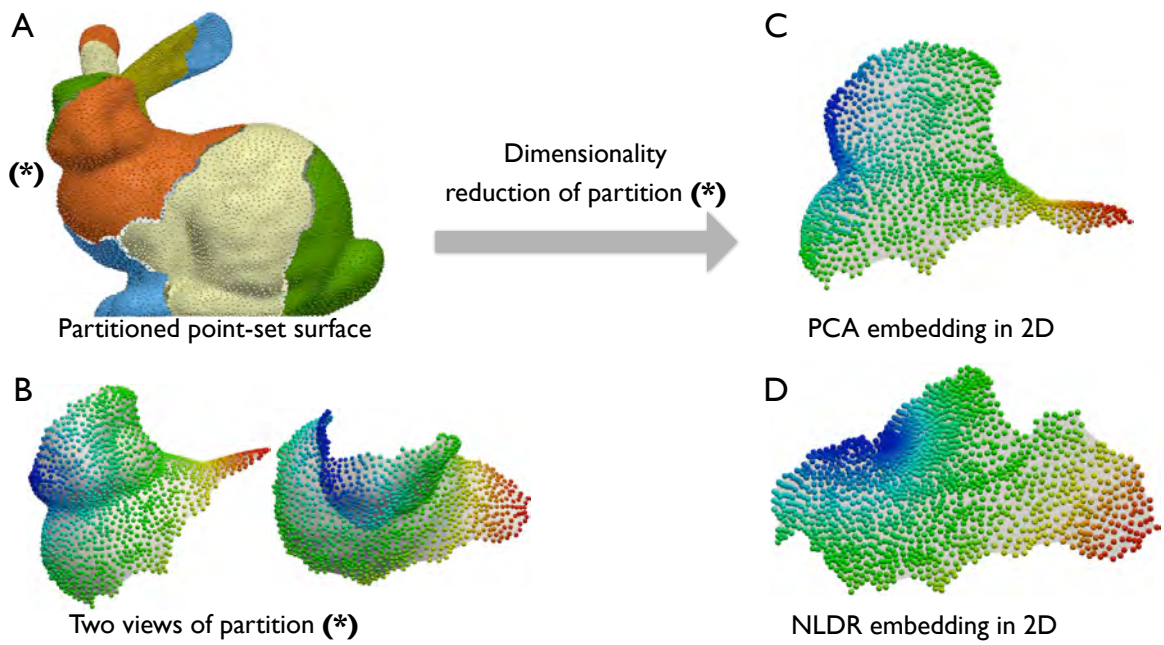


Figure 9: (A) The automatic partitioning of a point-set surface representing the Stanford bunny, done with METIS, can create patches of complex geometry and topology, e.g. a tubular partition in an ear. We recursively partition such patches. (B) Two views of the patch (*). Low-dimensional embedding of the patch (*) by PCA (C) and by a nonlinear method (D). The colors of the points are provided to guide the visual inspection of the embeddings of low-dimension. PCA collapses large regions of the patch, while the NLDR method successfully "irons" the curved patch into a moderately distorted low-dimensional embedding.

The low-dimensional space where the data points of a partition are embedded is a convenient parametric space for the corresponding patch. Let $p_a(\boldsymbol{\xi})$ denote the local *max-ent* approximants associated to the point-set Ξ_κ on a domain $\mathscr{A}_\kappa \subset \mathbb{R}^d$, a subset of the convex hull of the reduced node set conv Ξ_κ. We locally parametrize the manifold in this patch as

$$\begin{aligned} \boldsymbol{\varphi}_\kappa : \mathscr{A}_\kappa &\longrightarrow \mathbb{R}^D \\ \boldsymbol{\xi} &\longmapsto \sum_{a \in \mathcal{J}_\kappa} p_a(\boldsymbol{\xi})\, \boldsymbol{P}_a \end{aligned} \tag{2}$$

Note that the images of the local patches, $\boldsymbol{\varphi}_\kappa(\mathscr{A}_\kappa)$, overlap in the vicinity of the partition boundaries, and need not exactly coincide in these regions.

It is important to note that the embedded points are in general unstructured, and that, although until now we have shown examples in $d = 2$, the methodology is applicable to higher dimensional manifolds, as will became evident in the subsequents Sections and Appendices. Thus, a general method to process embedded smooth manifolds demands a smooth approximation scheme for general unstructured nodes in multiple dimensions. Here, we consider a general meshfree method to produce such approximants in any spacial dimension d, the local *max-ent* approximants developed by Arroyo and Ortiz (2006). Figure 10 illustrates this idea, and the general pipeline of the current method, for simplicity we have considered only one patch.

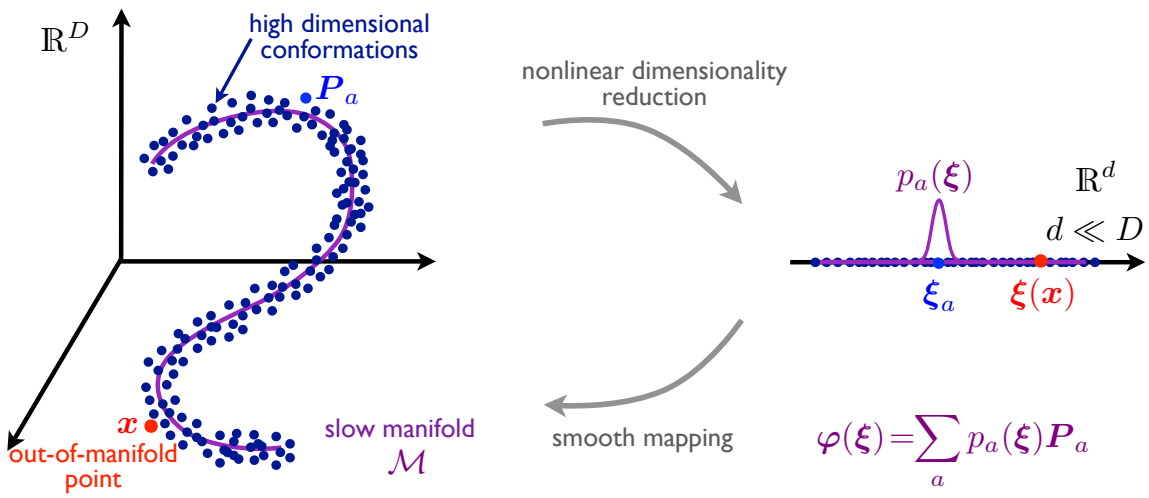


Figure 10: Sketch of the method to smoothly parametrize point-set manifolds. The set of high-dimensional snapshots X (dark blue points) are embedded in low dimensions (light-blue points) by NLDR methods, possibly applied to the pruned snapshots for computational convenience. With the low-dimensional embedding at hand, the slow manifold hidden in the training set X is represented with a smooth, data-driven parametrization $\boldsymbol{\varphi} : \mathscr{A} \subset \mathbb{R}^d \longrightarrow \mathbb{R}^D$. It is also possible to find the low-dimensional representation $\boldsymbol{\xi}(\boldsymbol{x}) \in \mathbb{R}^d$ of a new out-of-sample configuration $\boldsymbol{x} \in \mathbb{R}^D$, where $\boldsymbol{\xi}(\boldsymbol{x})$ is the pre-image by $\boldsymbol{\varphi}$ of the closest-point projection of $\boldsymbol{x}$ on $\mathcal{M}$, found solving a nonlinear least-squares problem Millán et al. (2012).

Local *max–ent* approximants

The smooth approximants $p_a(\boldsymbol{\xi})$ used here are local *max-ent* (LME) meshfree approximants (Arroyo and Ortiz, 2006). These approximation schemes fall into the class of convex approximants, like natural neighbor approximants (Sukumar, 2004), subdivision approximants (Cirak et al., 2000), or Non-Uniform Rational B-splines (NURBS) and B-splines basis functions (Hughes et al., 2005).

The main features of LME approximants are the following:

- The shape functions are strictly positive and C^∞.
- A weak version of the Kronecker-delta property is satisfied *ab initio* on the convex hull of the domain. This property makes the imposition of essential boundary conditions in the numerical approximation of first-order partial differential equations straightforward. This is a significant difference when LME basis functions are compared with most of the meshfree methods, in particular those based on the Moving Least Squares (MLS) approximation (Fernández-Méndez and Huerta, 2004).
- They are defined and efficiently computed in any dimension.
- They are non-interpolating in general, and satisfy up to the first-order consistency condition, although can be extended to second-order consistency (Cyron et al., 2010; González et al., 2010; Rosolen et al., 2012).
- Formally, LME shape functions have global support. However from a numerical perspective the shape functions decay as $\exp(-\beta\, r^2)$ with distance r to the corresponding node, which leads for all practical purposes to compactly supported functions.
- They are computed at each evaluation point through a strictly convex, feasible optimization problem, efficiently and robustly solved by duality.

- In Galerkin methods, they lead to very accurate solutions for smooth problems.
- They deal robustly with training data infected with zero-mean additive noise, as has been proven for nonparametric supervised classification (Gupta, 2003).

LME approximants stem from an optimal compromise between two competing objectives: (1) maximum locality of the shape functions (maximum correlation between the approximation and the nodal value at the closest points) and (2) Jaynes' information optimality, as stated by the principle of maximum entropy. The second objective is a statement of information-theoretical optimality in the sense that it provides the least biased approximation scheme consistent with the reproducing conditions. It is worth noting that in the formulation of Arroyo and Ortiz (2006), the support width for LME shape functions is modulated with a locality parameter $\beta(\boldsymbol{\xi}) > 0$, which also controls their decay.

For highly non-uniform node sets, basis functions of uniform aspect ratio require a highly non-uniform locality parameter $\beta(\boldsymbol{\xi})$. Defining such a function, whose smoothness limits that of the basis functions, is not easy in general. To beat this difficulty, we adopt a slightly different approach presented by Rosolen et al. (2010), by which the locality is a nodal attribute, β_a with $a = 1, \dots, n$, rather than a continuous function (see Box 2). It is more informative to characterize the locality of the basis functions with the dimensionless aspect ratio parameter $\gamma_a = \beta_a h_a^2$, where h_a is the typical nodal spacing in the vicinity of $\boldsymbol{\xi}_a$. As the value of γ_a increases, the corresponding shape function is sharper and more local. In the limit of all $\gamma_a \to \infty$, we recover the Delaunay affine shape functions. For highly unstructured point-sets, it is easy to produce local *max-ent* approximants of uniform aspect ratio by appropriately selecting a non-uniform value of the parameters β_a, resulting in a uniform value for the parameters γ_a. The expressions for the first and second spatial derivatives of local *max-ent* approximants have been obtained by Rosolen et al. (2010) and Millán et al. (2011), respectively. The local *max-ent* basis functions in two dimensions are illustrated in Fig. 11.

In the LME convex program, the entropy and locality maximization are competing objectives. Following Arroyo and Ortiz (2006), a standard device to harmonize competing objectives in multi-criterion optimization is to seek for Pareto optimal points. In the present context, the Pareto set is the set of convex approximants such that no other convex approximant is better or dominates. Here we consider a node-wise weighting of the locality objective (Rosolen et al., 2010), leading to

$$\begin{aligned} \text{(LME)} \quad \text{For fixed } \boldsymbol{x} \text{ minimize} \quad & f_\beta(\boldsymbol{\xi}, \boldsymbol{p}) \equiv U_\beta(\boldsymbol{\xi}, \boldsymbol{p}) - H(\boldsymbol{p}) \\ \text{subject to} \quad & p_a \geq 0, \quad a = 1, \dots, n \\ & \sum_{a=1}^{n} p_a = 1, \quad \sum_{a=1}^{n} p_a \boldsymbol{\xi}_a = \boldsymbol{\xi}, \end{aligned}$$

where $\boldsymbol{p} = \{p_a\}_{a=1,\dots,n} \in \mathbb{R}^n_+$ are the local-max approximation schemes, while the *entropy* $H(\boldsymbol{p}) = -\sum_a p_a \ln p_a$ and the *locality* $U_\beta(\boldsymbol{\xi}, \boldsymbol{p}) = \sum_a \beta_a p_a |\boldsymbol{\xi} - \boldsymbol{\xi}_a|^2$ are the competing objective functions to be minimized. The three remainder constraints encode the *positivity* of the shape functions, and the *zeroth-* and *first-order* consistency conditions.

This convex optimization problem is guaranteed to be solvable within the convex hull of the node set for any non-negative β_a. Standard duality methods provide an effective method to solve the nonlinear, convex optimization program in (LME) (Arroyo and Ortiz, 2006).

Box 2: Convex optimization program defining the local *max-ent* approximants.

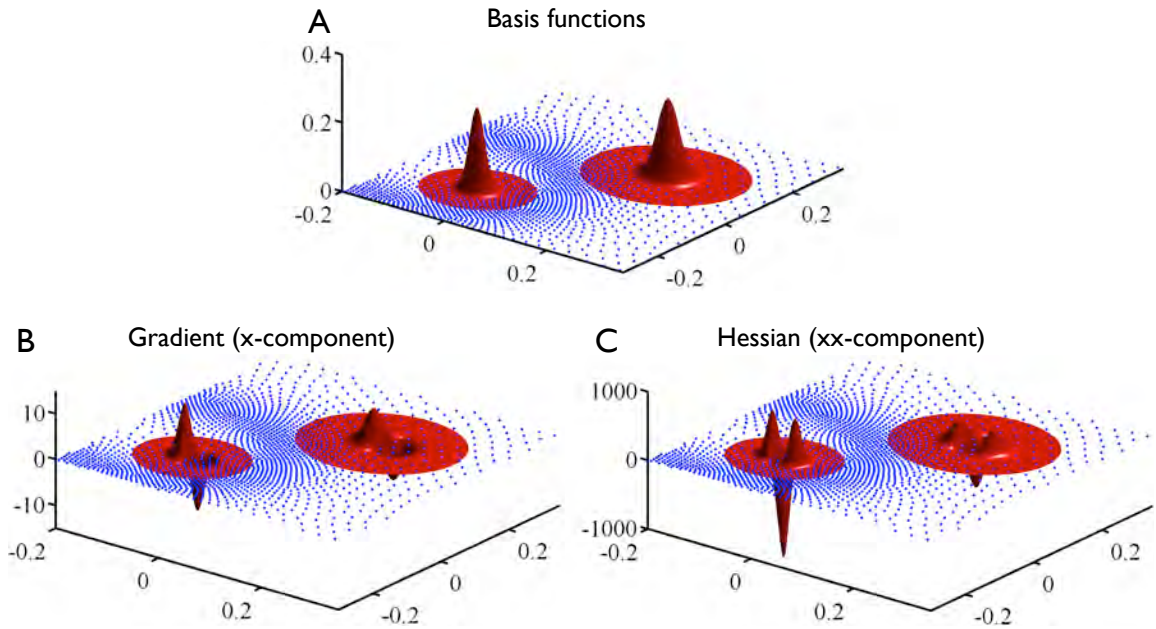


Figure 11: (A) Local *max-ent* basis functions computed with an aspect ratio parameter $\gamma = 0.8$ in an unstructured and non-uniform two-dimensional distribution of points. (B-C) The $x-$components of the gradient and the Hessian.

3.4 Partition of unity to evaluate integrals on $\mathcal{M}$

Consider for instance the integral of a scalar function f over a manifold $\mathcal{M}$, $f : \mathcal{M} \to \mathbb{R}$. Then we have the following identity

$$\int_{\mathcal{M}} f(\boldsymbol{x})\, d\mathcal{M} = \sum_{\kappa=1}^{L} \int_{\mathcal{M}} \psi_\kappa(\boldsymbol{x})\, f(\boldsymbol{x})\, d\mathcal{M}.$$

Combining the *partition of unity* with the *local parametrization* of the κ-th patch, $\boldsymbol{\varphi}_\kappa$, we can approximate numerically integrals over the manifold $\mathcal{M}$ described by a set of scattered points as

$$\int_{\mathcal{M}} f(\boldsymbol{x})\, d\mathcal{M} \simeq \sum_{\kappa=1}^{L} \int_{\mathscr{A}_\kappa} \psi_\kappa(\boldsymbol{\varphi}_\kappa(\boldsymbol{\xi}))\, f(\boldsymbol{\varphi}_\kappa(\boldsymbol{\xi}))\, J_\kappa(\boldsymbol{\xi})\, d\boldsymbol{\xi}$$

where $J_\kappa = \sqrt{\det\left[(D\boldsymbol{\varphi}_\kappa)^T D\boldsymbol{\varphi}_\kappa\right]}$ is the Jacobian determinant of the parametrization. In this way, similarly to finite element methods, we have split the integral into local contributions, which can be evaluated using local parametrizations. The last integral can be subsequently approximated by numerical quadrature on the local parametric space.

3.5 Single representation of a manifold described by multiple patches

We have avoided so far a precise definition of a numerical surface in the overlapping regions. Although a single representation of the manifold given by multiple overlapping patches, which do not coincide exactly at the overlap regions, is not needed to compute integrals and functionals on the manifold, such a representation is useful in a number of situations such as visualization, contact detection,

or imposition of displacements at interior points. We describe now how we proceed. As a starting point, we have an out-of-sample point $\boldsymbol{x} \in \mathbb{R}^D$ in the vicinity of the embedded manifold. This point could be a sampling of the actual manifold, or the image of a point in parameter space by a patch parametrization. Our goal is to define an operator mapping $\boldsymbol{x}$ onto the manifold. We first identify the patches that have an influence on $\boldsymbol{x}$ by building the index set

$$\mathcal{N}_{\boldsymbol{x}} = \{\kappa \in \{1, 2, \ldots, L\} \mid \psi_\kappa(\boldsymbol{x}) > \texttt{TOL}\}.$$

We can then find the preimage of the closest point projection of $\boldsymbol{x}$ onto the multiple patch representations, which we denote by $\boldsymbol{\xi}_\kappa(\boldsymbol{x}) = \boldsymbol{\varphi}_\kappa^{-1}(\boldsymbol{\pi}_\kappa(\boldsymbol{x}))$ for $\kappa \in \mathcal{N}_{\boldsymbol{x}}$, see Appendix B. We can then define the following operator (almost a projection)

$$\boldsymbol{\pi}(\boldsymbol{x}) = \sum_{\kappa \in \mathcal{N}_{\boldsymbol{x}}} \psi_\kappa(\boldsymbol{x})\ \boldsymbol{\varphi}_\kappa(\boldsymbol{\xi}_\kappa(\boldsymbol{x})), \tag{3}$$

which averages the position of the point as represented by the different overlapping parametrizations. Similarly to the definition of MLS point-set surfaces (Levin, 2003; Alexa et al., 2001), we can formally define the numerical surface at overlapping regions as the fixed points of this operator. For MLS point-set surfaces, the properties of the fixed points of a related operator have been mathematically analyzed, and the properties of the resulting manifold established. Our numerical experiments indicate that successive iterations of $\boldsymbol{\pi}(\boldsymbol{x})$ converge extremely fast. In practice, we do not iterate the operator.

Contributions

We summarize here the most significant contributions of this work, which is the backbone of the papers of Millán et al. (2011) and Millán et al. (2012):

1. A methodology to handle non-uniformly sampled point-set manifolds has been proposed. The core of the method is the partition of the manifold into patches, which are embedded in low-dimension through manifold learning techniques, and then smoothly parametrized.

2. The standard methods PCA, MLLE and ISOMAP, and new variations, IAC and NLIE, have been developed, implemented and tested in a variety of examples in Matlab and in C++ languages.

3. An automatic recursive partitioning algorithm has been designed. It has been exhaustively and successfully applied to deal with general point-set surfaces.

4. We have implemented LME basis functions and their first and second spatial derivatives in an efficient and compact way for any dimension, both in Matlab and in C++ languages.

5. Giben a new out-of-sample point $\boldsymbol{x} \in \mathbb{R}^D$, an iterative nonlinear algorithm has been developed to compute its low-dimensional representation, $\boldsymbol{\xi}(\boldsymbol{x}) \in \mathbb{R}^d$, and its closest point projection $\boldsymbol{\pi}(\boldsymbol{x}) \in \mathbb{R}^D$.

Open lines for research

We next discuss some open ideas for research derived from this work:

(i) Improvement of the partitioning algorithm in any dimension. This extension should be easy to implement because the core operations are done with a proximity graph by METIS library (Karypis and Kumar, 1998).

(ii) Mathematical properties of the operator $\boldsymbol{\pi}(\boldsymbol{x})$ and its iteration.

(iii) The NLDR methods try to capture the geometrical features of the manifold hidden in high dimensional data. In specific applications, these methods are completely blind to the underlying physics behind the data. We envision to include such physical information in the computation of the low-dimensional embeddings. For instance, in the study of molecular conformations, the Boltzman distribution could be used to define better-suited similarity metrics, involved in all NLDR methods.

4 Meshfree thin-shell analysis

In this section, we review the mechanics of thin shells, gently exposed in the works of Cirak et al. (2000) and Cirak and Ortiz (2001), which are based on the geometrically exact formulation derived by Simo and Fox (1989). We restrict our attention to the Kirchhoff–Love theory of thin shells.

4.1 Introduction

The Kirchhoff–Love theory of thin shells is very attractive as compared to shear deformable shells since only the middle surface (three degrees of freedom per node) needs to be described. It has been shown to be a very good model for slender surface-like bodies. However, this theory involves both the first and the second fundamental forms of the surface, and consequently the approximation of the deformation needs to have second order square integrable derivatives. For general unstructured meshes, it is difficult to define C^1 finite element approximations, which has prompted a myriad of techniques that avoid this requirement, see MacNeal and Harder (1985); Bucalem and Bathe (1993); Simo and Fox (1989) and references there in. Yet, C^0 approaches do not perform well in the thin shell limit.

Early meshfree approaches are amongst the first numerical methods with smooth approximants for Kirchhoff–Love shells beyond Hermite approximations (Krysl and Belytschko, 1996). Following ideas from computer graphics, smooth subdivision surfaces finite elements have been proposed for thin shells (Cirak et al., 2000; Cirak and Ortiz, 2001). Subdivision finite elements provide a unified framework for geometric modeling and thin shell analysis. Along this line of work, Isogeometric analysis (Hughes et al., 2005; Cottrell et al., 2009) is a new technology building on Computer Aided Design (CAD) smooth approximation methods, such as B-Splines and NURBS. They have been successfully applied to beams, plates, and thin shells (Cottrell et al., 2006, 2009; Kiendl et al., 2009). These methods exhibit a high fidelity of the geometry representation, and their difficulties derived from the rigidity of the NURBS framework with regards to the structure of the grid are the topic of current research (Bazilevs et al., 2010). On the other hand, Discontinuous Galerkin (DG) formulations have been proposed recently for plates, beams and thin shells (Engel et al., 2002; Wells and Dung, 2007, 2008; Noels and Radovitzky, 2008; Noels, 2009). These methods avoid the C^1 continuity requirement by designing suitable numerical fluxes conjugate to the deformation jumps. An advantage of this method is the ease in the imposition of the rotation essential boundary conditions. As disadvantages, the formulation and implementation of these methods is cumbersome, and they typically exhibit a poorer accuracy for a given number of degrees of freedom as compared to methods based on smooth approximants.

Over the last years, there has been a growing interest in the computer graphics community on point-based surface processing, which presents attractive features as compared to conventional mesh-based processing (Alexa et al., 2001; Pauly, 2003). In mesh-based methods, the mesh serves two useful purposes: it describes the geometry of the surface, and the elements provide local parametric spaces where the shape functions and the local parametrizations of the surface can be defined, and where the required calculations on the surface can be performed, e.g. for thin shell analysis. In these methods, the mesh generation can be difficult, they are not natural for point-based data, and they seem unpractical for embedded manifolds in high dimensional spaces. On the other hand, in the absence of a mesh, the notion of a surface defined from a set of scattered nodes becomes difficult to grasp. Indeed, unlike functional approximation in Euclidean space, in the case of embedded surfaces there is in general no single parametric space in which the shape functions can be defined. This issue is nicely put by Levin (2003).

In the computer graphics literature, Levoy and Whitted (1985) pioneered using points as primitives for geometric modeling and rendering of surfaces. Existing methods for describing a surface from a set of scattered points are generally based on implicit representations of the surface (Hoppe

et al., 1992; Ohtake et al., 2003). MLS surfaces are a noteworthy example of point-based implicit surface representation, where the surface is defined as the set of fixed points of suitable projections (Levin, 1998, 2003). This idea has been very successful in the computer graphics community for rendering, either up or down sampling, and manipulating point-set models, e.g. see the works by Alexa et al. (2001); Pauly (2003); Alexa et al. (2003); Amenta and Kil (2004); Alexa et al. (2004). Despite the common themes and challenges, these developments have remained largely unconnected to the computational mechanics community. As an example, meshfree methods have been applied to thin-shell analysis, and the difficulty of defining an appropriate parametric space has been overcome by considering either a support mesh or very simple surfaces admitting a single parametric space (Krysl and Belytschko, 1996; Noguchi et al., 2000; Chen and Wang, 2006; Rabczuk et al., 2007).

In the present work we greatly expanded the range of applicability of meshfree thin shell analysis, through automatic, efficient and robust local parametrizations of the point-set surface. We exploit NLDR in conjunction with the automatic partitioning of the point-set surface, which exhibit a small overlap, see Appendix B.

4.2 Kinematics of the shell

We next describe the kinematics of a thin shell body $\mathscr{S} \subset$ in three-space. We assume that this body can be described by the pair $(\boldsymbol{\varphi}, \boldsymbol{t})$, where the mapping $\boldsymbol{\varphi}$ defines the shell middle surface, Ω, and $\boldsymbol{t}$ is a field of unit vectors (a field of directors). We assume the thickness h of the shell to be uniform for simplicity, and also we assume that the change in shell thickness after deformation is negligible. We follow the usual convention for Latin and Greek indices (i.e. $i = 1, 2, 3$; $\alpha = 1, 2$), a comma denotes partial differentiation, subscripts refer to covariant components, and superscripts denote contravariant components. Then, the thin shell body $\mathscr{S}$ is given by

$$\mathscr{S} = \left\{ \boldsymbol{\Phi} \in \mathbb{R}^3 | \ \boldsymbol{\Phi} = \boldsymbol{\varphi}(\xi^\alpha) + \xi \ \boldsymbol{t}(\xi^\alpha), \ \ -\frac{h}{2} \leq \xi \leq \frac{h}{2}, \ (\xi^1, \xi^2) \in \mathscr{A} \right\},$$

where $\mathscr{A} \subset \mathbb{R}^2$ is the parametric space for the middle surface (see Fig. 12). Hence, we view a configuration $\boldsymbol{\Phi}$ as a mapping from a parametric domain $\mathscr{A} \times [-h/2, h/2]$ into $\mathbb{R}^3$. The parametric domain is described by the coordinates $\{\xi^1, \xi^2, \xi^3\}$ (where we identify $\xi = \xi^3$), whose corresponding dual basis is $\{\boldsymbol{E}^i\}$. The area element of the middle surface can be computed as $d\Omega = \bar{j} \ d\xi^1 d\xi^2$, where $\bar{j} = \left| \boldsymbol{\varphi}_{,1} \times \boldsymbol{\varphi}_{,2} \right|$. The tangent map of a given configuration $T\boldsymbol{\Phi}$ can be computed from the convective basis vectors $\boldsymbol{g}_i$ as

$$T\boldsymbol{\Phi} = \frac{\partial \boldsymbol{\Phi}}{\partial \xi^i} \otimes \boldsymbol{E}^i = \boldsymbol{g}_i \otimes \boldsymbol{E}^i,$$

with $\boldsymbol{g}_\alpha = \frac{\partial \boldsymbol{\Phi}}{\partial \xi^\alpha} = \boldsymbol{\varphi}_{,\alpha} + \xi \ \boldsymbol{t}_{,\alpha}$ and $\boldsymbol{g}_3 = \frac{\partial \boldsymbol{\Phi}}{\partial \xi} = \boldsymbol{t}$. The covariant components of the metric tensor in convected coordinates are given by $g_{ij} = \boldsymbol{g}_i \cdot \boldsymbol{g}_j$.

Hereinafter, subscript 0 denotes quantities in the reference configuration, for instance $\boldsymbol{\varphi}_0$ is a point on the reference middle surface. A deformation mapping is a mapping from a reference body into $\mathbb{R}^3$, $\boldsymbol{\Phi} \circ \boldsymbol{\Phi}_0^{-1}$. Consequently, the deformation gradient is $\boldsymbol{F} = T\boldsymbol{\Phi} \ (T\boldsymbol{\Phi}_0)^{-1}$, and the Jacobian is $J = \det(\boldsymbol{F}) = j/j_0$, where $j = \det(T\boldsymbol{\Phi}) = \boldsymbol{g}_3 \cdot (\boldsymbol{g}_1 \times \boldsymbol{g}_2)$.

The shell director in the reference configuration $\boldsymbol{t}_0$ coincides with the normal to the undeformed middle surface of the shell and hence has the properties

$$\boldsymbol{t}_0 = \frac{\boldsymbol{\varphi}_{0,1} \times \boldsymbol{\varphi}_{0,2}}{\bar{j}_0}, \quad \boldsymbol{\varphi}_{0,\alpha} \cdot \boldsymbol{t}_0 = 0, \quad |\boldsymbol{t}_0| = 1, \quad \boldsymbol{t}_0 \cdot \boldsymbol{t}_{0,\alpha} = 0.$$

The local shell deformations can be characterized by the Green–Lagrange strain tensor expressed as the difference between the metric tensors on the deformed and undeformed configurations of the

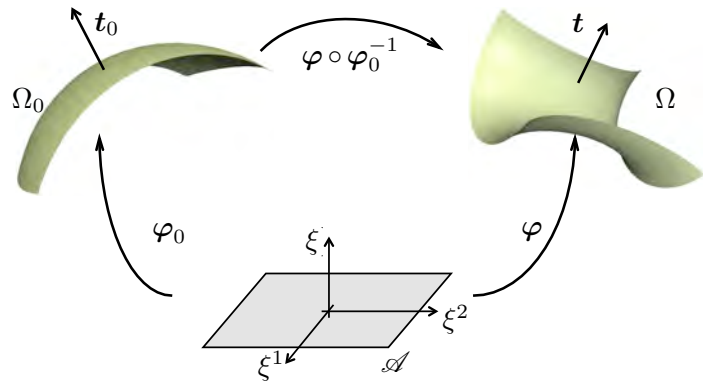


Figure 12: Schematic description of the reference, deformed and parametric configurations of the middle shell surface.

shell, i.e.

$$E_{ij} = \frac{1}{2}\left(g_{ij} - g_{0ij}\right) = \frac{1}{2}(\boldsymbol{\Phi}_{,i} \cdot \boldsymbol{\Phi}_{,j} - \boldsymbol{\Phi}_{0,i} \cdot \boldsymbol{\Phi}_{0,j}).$$

Plugging the basic kinematic ansatz $\boldsymbol{\Phi} = \boldsymbol{\varphi}(\xi^\alpha) + \xi\ \boldsymbol{t}(\xi^\alpha)$ into the above expression, and grouping terms, we obtain

$$E_{ij} = \varepsilon_{ij} + \xi\ \rho_{ij} + (\xi)^2\ \vartheta_{ij},$$

which admits the following interpretation in terms of the symmetric tensors ε_{ij}, ρ_{ij} and ϑ_{ij}:

- The membrane strain tensor $\varepsilon_{\alpha\beta} = \frac{1}{2}(\boldsymbol{\varphi}_{,\alpha} \cdot \boldsymbol{\varphi}_{,\beta} - \boldsymbol{\varphi}_{0,\alpha} \cdot \boldsymbol{\varphi}_{,\beta})$, which lives on the middle surface, measures the in-plane deformation of the surface; the components $\varepsilon_{\alpha 3} = \frac{1}{2}\boldsymbol{\varphi}_{,\alpha} \cdot \boldsymbol{t}$ measure the shearing of the director $\boldsymbol{t}_0$; and the component $\varepsilon_{33} = \frac{1}{2}(\boldsymbol{t} \cdot \boldsymbol{t} - 1)$ measures the stretching of the director $\boldsymbol{t}_0$.

- The bending or change in curvature of the shell is measured by the tensor $\rho_{\alpha\beta} = \boldsymbol{\varphi}_{,\alpha} \cdot \boldsymbol{t}_{,\beta} - \boldsymbol{\varphi}_{0,\alpha} \cdot \boldsymbol{t}_{0,\beta}$, and $\rho_{\alpha 3} = \frac{1}{2}\boldsymbol{t}_{,\alpha} \cdot \boldsymbol{t}$ measures the shearing originated from the director elongation; the in-plane tensor $\vartheta_{\alpha\beta} = \frac{1}{2}(\boldsymbol{t}_{,\alpha} \cdot \boldsymbol{t}_{,\beta} - \boldsymbol{t}_{0,\alpha} \cdot \boldsymbol{t}_{0,\beta})$ is exclusively related to changes of the middle surface directors. The rest of the components vanish, $\rho_{33} = \vartheta_{3i} = \vartheta_{i3} = 0$.

Kirchhoff–Love hypothesis

In the remainder of this section we restrict our attention to the Kirchhoff–Love theory of thin shells, i.e. we constrain the deformed director $\boldsymbol{t}$ to coincide with the unit normal of the deformed middle surface of the shell, i.e.

$$\boldsymbol{t} = \frac{\boldsymbol{\varphi}_{,1} \times \boldsymbol{\varphi}_{,2}}{\bar{j}}, \quad \boldsymbol{\varphi}_{,\alpha} \cdot \boldsymbol{t} = 0, \quad |\boldsymbol{t}| = 1, \quad \boldsymbol{t} \cdot \boldsymbol{t}_{,\alpha} = 0.$$

This assumption is well suited when the ratio between the shell thickness and its characteristic size is $\ll 1$. Kirchhoff–Love kinematic assumption generates a formulation of the shell exclusively in terms of the shell middle surface such that the terms related with the shear and the stretching in the normal direction are identical to zero. Therefore, the only remaining non-zero components of the Green–Lagrange strain tensor are

$$E_{\alpha\beta} = \varepsilon_{\alpha\beta} + \xi\ \rho_{\alpha\beta} + (\xi)^2\ \vartheta_{\alpha\beta}. \tag{4}$$

Equilibrium configurations

The potential energy of an elastic shell body with bulk internal energy density W can be expressed as

$$\Pi[\boldsymbol{\Phi}] = \int_{\mathscr{S}_0} W(E_{ij})\ dV_0 + \Pi_{\mathtt{ext}}[\boldsymbol{\Phi}],$$

where $\Pi_{\mathtt{ext}}$ is the potential energy of the external loads. For concreteness, we consider an isotropic Kirchhoff–Saint Venant elastic material (Ciarlet, 2000)

$$W = \frac{1}{2} C^{ijkl} E_{ij} E_{kl},$$

where C^{ijkl} are the contravariant components of the elasticity tensor.

For thin shell bodies, the Green-Lagrange tensor components are commonly retained up to first order in h, see Eq. (4), and the effect of curvature on the Jacobian away from the middle surface is neglected, that is $j_0/\bar{j}_0 = 1$ (see Simo and Fox (1989); Simo et al. (1989)). Assuming that the elasticity tensor does not vary trough the thickness, the internal energy density can be integrated through-the-thickness, resulting in an internal energy density per unit area

$$\mathcal{W}(E_{\alpha\beta}) = \frac{1}{2}\int_{-h/2}^{h/2} C^{\alpha\beta\gamma\delta} E_{\alpha\beta} E_{\gamma\delta} \frac{j_0}{\bar{j}_0}\ d\xi \simeq \frac{1}{2} C^{\alpha\beta\gamma\delta} \left(h\ \varepsilon_{\alpha\beta}\varepsilon_{\gamma\delta} + \frac{h^3}{12}\rho_{\alpha\beta}\rho_{\gamma\delta} \right).$$

Thus, the internal potential energy is a functional of the middle surface configuration, which can be written as an integral over the reference middle surface

$$\Pi_{\mathtt{int}}[\boldsymbol{\varphi}] = \int_{\Omega_0} \mathcal{W}(E_{\alpha\beta})\ d\Omega_0,$$

and the external potential becomes

$$\Pi_{\mathtt{ext}}[\boldsymbol{\varphi}] = -\int_{\Omega_0} \boldsymbol{q}\cdot\boldsymbol{\varphi}\ d\Omega_0 - \int_{\partial\Omega_0} \boldsymbol{h}\cdot\boldsymbol{\varphi}\ d\ell_0,$$

where $\boldsymbol{q}$ is the external body load per unit area, $\boldsymbol{h}$ the forces per unit length applied on the boundary of the middle surface, and $d\ell_0$ is the line element of the boundary of the middle surface. Distributed torques can be also applied at the boundary of the thin shell.

Following Simo and Fox (1989), we introduce the elastic constitutive relations between the shell stresses and the strains as

$$\begin{aligned} n^{\alpha\beta} &= \frac{\partial \mathcal{W}}{\partial \varepsilon_{\alpha\beta}} = h\ C^{\alpha\beta\gamma\delta}\varepsilon_{\gamma\delta}, \\ m^{\alpha\beta} &= \frac{\partial \mathcal{W}}{\partial \rho_{\alpha\beta}} = \frac{h^3}{12} C^{\alpha\beta\gamma\delta}\rho_{\gamma\delta}, \end{aligned}$$

where $n^{\alpha\beta}$ is the membrane stress resultant and $m^{\alpha\beta}$ is the bending stress resultant.

The stable equilibrium configurations of the shell minimize the total potential energy, subject to the boundary conditions, and consequently, satisfy the principle of virtual work, expressed here in terms of integrals over the parametric space $\mathscr{A}$

$$0 = \delta\Pi[\boldsymbol{\varphi}, \delta\boldsymbol{\varphi}] = \int_{\mathscr{A}} (\delta\boldsymbol{\varepsilon}\cdot\boldsymbol{n} + \delta\boldsymbol{\rho}\cdot\boldsymbol{m})\, \bar{j}_0\ d\xi^1 d\xi^2 + \delta\Pi_{\mathtt{ext}}[\delta\boldsymbol{\varphi}].$$

4.3 Ritz-Galerkin discretization

We consider now the discrete equilibrium equations for a shell whose middle surface in the reference configuration is numerically represented with the procedure described before, in terms of a set of nodes $P_0 = \{\boldsymbol{P}_{01}, \dots, \boldsymbol{P}_{0N}\}$, and a set of L patches. We follow a total Lagrangian approach, with the same parameter space and basis functions for the reference and deformed configurations. Let $\boldsymbol{\varphi}_{0\kappa}$ be the reference configuration mapping for the middle surface of a specific patch κ, defined over the parametric space $\mathscr{A}_\kappa$

$$\boldsymbol{\varphi}_{0\kappa}(\boldsymbol{\xi}) = \sum_{a \in \mathcal{J}_\kappa} p_a(\boldsymbol{\xi})\ \boldsymbol{P}_{0a},$$

as described in Section 3.3. We represent the deformed configuration in a given patch κ as

$$\boldsymbol{\varphi}_{\kappa}(\boldsymbol{\xi}) = \sum_{a \in \mathcal{J}_\kappa} p_a(\boldsymbol{\xi})\ \boldsymbol{P}_{a}.$$

With the strategy presented in Section 3.4 and the above definitions, the internal elastic energy of the discretized shell can be split into patch contributions

$$\Pi^h_{\mathtt{int}}(\boldsymbol{P}_1, \boldsymbol{P}_2, \dots, \boldsymbol{P}_N) = \sum_{\kappa=1}^{L} \int_{\mathscr{A}_\kappa} (\psi_\kappa \circ \boldsymbol{\varphi}_{0\kappa})\ \mathcal{W}(E_{\alpha\beta})\ \bar{j}_0\ d\xi^1 d\xi^2,$$

where $E_{\alpha\beta}$ and $\bar{j}_0$ are evaluated with the κ-th patch approximation of the undeformed and deformed configurations. Note that only $E_{\alpha\beta}$ depends on the unknown control points defining the deformed configuration. The external potential is numerically computed likewise. Equilibrium configurations satisfy that the out-of-balance forces vanish

$$0 = \frac{\partial \Pi^h}{\partial \boldsymbol{P}_a}(P) = \boldsymbol{f}^a_{\mathtt{int}}(P) - \boldsymbol{f}^a_{\mathtt{ext}}(P).$$

Stable equilibrium configurations are obtained by numerically minimizing $\Pi^h(P)$, where the essential displacement and rotation boundary conditions are imposed with Lagrange multipliers in an augmented Lagrangian framework. Within the augmented Lagrangian loop, we first obtain a coarse and robust approximation of the equilibrium point with a limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method, and then switch to Newton's method combined with line-search to refine the minimization. Details about the calculation of the out-of-balance forces, the tangent stiffness matrix, the boundary constraints, and the solution method are given in Appendices B and E2.

4.4 Numerical examples

We exercise the proposed method treating small deformations in the classical benchmark of a hemisphere loaded with two pairs of facing concentrated forces Belytschko et al. (1985). Then, we show the flexibility of the proposed methodology dealing with shells of complex topology undergoing large deformations. A collection of videos highlighting the nonlinear mechanics of these geometrically exact shells have been given by Millán (2011).

Numerical aspects

For a detailed account about the definition of the LME basis functions and the numerical parameters involved, see Millán et al. (2011, 2012). We note here that linearly reproducing local LME approximants with relatively wide support can very accurately approximate thin shell problems with

functionals involving second order derivatives. The smoothness or aspect ratio of the basis functions is controlled by a non-dimensional parameter. We choose $\gamma_{LME} = 0.8$, which provides accurate solutions at a moderate computational cost. Similarly, for the partition of unity Shepard functions, we select $\gamma_{PU} = 4.0$, which results in moderately narrow overlap regions.

Pinched closed hemisphere

In this example, a hemispherical shell of radius $R = 10$ and thickness $h = 0.04$ is subjected to two pairs of radial loads $F = 2$ acting along diametral directions, see Fig. 13. This is a challenging test, which assesses the method's ability to represent inextensional deformations under complex shell bending conditions with curvature in two directions. The convergence of the relative error for the radial displacement is shown in Fig. 13. The displacements are normalized by a deflection of δ_r =0.09241 obtained by an overkill calculation, which agrees with the lower bound given by MacNeal and Harder (1985); Belytschko et al. (1985), that is δ_r =0.0924. In this figure, we plot the convergence results reported by Green and Turkiyyah (2004) for subdivision finite elements based on Loop's scheme (triangular elements) and on Catmull-Clark's scheme (quadrilateral elements), as well as results with the previous version of our method (Millán et al., 2011). The excellent convergence properties of the proposed method is clear from the figure. We obtain more accurate results for a given number of degrees of freedom than arguably the most competitive method for thin shells. Our method is more expensive than subdivision finite elements due to the quadrature and the larger sparsity pattern of the stiffness matrix, which makes a full comparison difficult. Also, we can see that the results given by the present method, that uses MLLE, are very similar to those obtained with a local linear manifold learning method, which is much more expensive (Millán et al., 2012). Finally, we note the effect of fitting the control points to better represent the geometry. The control and patch points and the deformation are shown in Fig. 14.

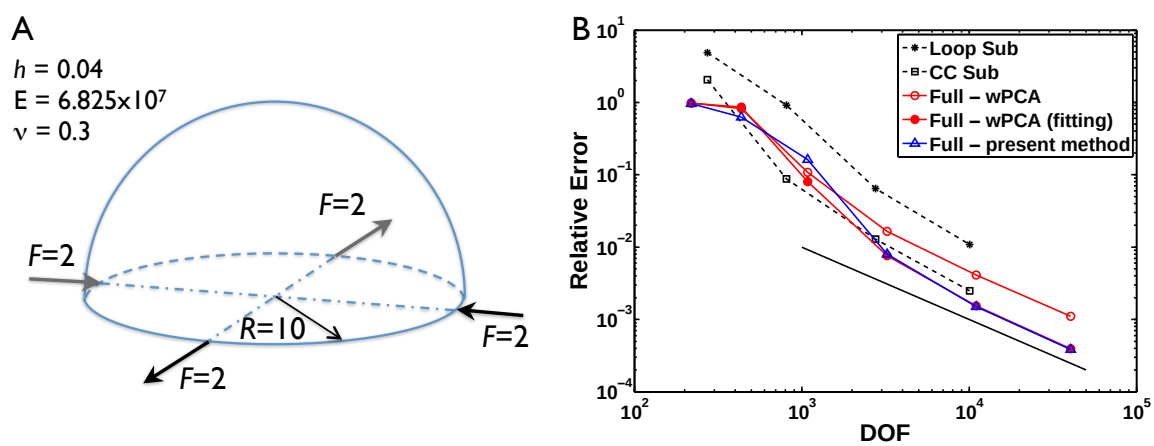


Figure 13: (A) Sketch of the pinched closed hemisphere shell test. (B) Convergence of the normalized radial displacement for two subdivision schemes (Green and Turkiyyah, 2004), for the weithed-PCA method proposed in Millán et al. (2011), and for the present method based on MLLE.

Connected pipes

We now illustrate the ability of the proposed method to deal with extremely complex topologies, see Fig. 15. The boundary curves of the bottom pipes are clamped and the top boundary curves are incrementally displaced in the $(-1, -1, -1)$ direction. The lateral dimension of the system is

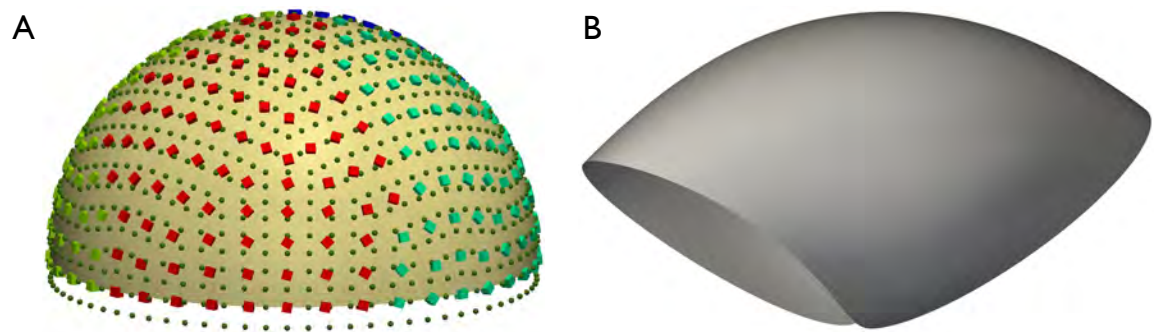


Figure 14: Pinched hemisphere: geometric markers ⋄, control points ∘, and reference configuration (A), deformed surface (B). In this linear problem, the deformation has been amplified by a factor of one hundred.

6.2 and the shell thickness is $h = 0.03$. Figure 15 illustrates the discretization with 14176 control points. Figure 16 shows the energy density on the deformed configuration at four snapshots, without magnification of the displacements. The shell undergoes several geometric instabilities and exhibits localized elastic deformations. We insist on the fact that this thin shell cannot be studied with previous meshfree approaches because it does not admit a single parameter space.

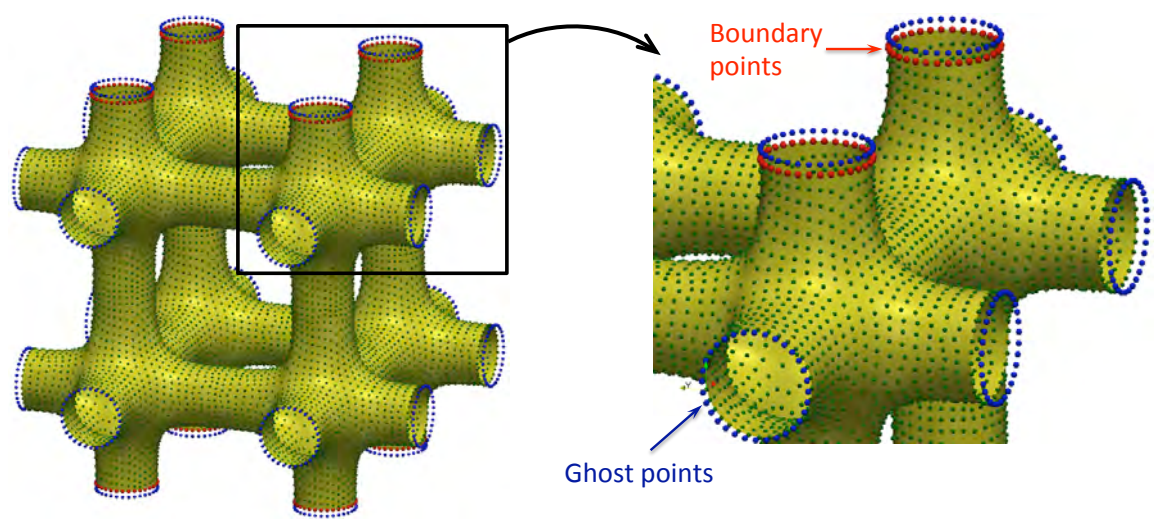


Figure 15: Point-set surface for the example of the connected pipes. The control points where the boundary conditions are applied, and the ghost control points are highlighted. The boundary curves of the bottom pipes are clamped and the top boundary curves are incrementally displaced in the downward diagonal $(-1, -1, -1)$ direction.

Contributions

We summarize here the most significant contributions of this work:

1. We have exercised our methodology to process point-set manifolds in Kirchhoff-Love thin shell analysis. The performance of the method, assessed by the classical obstacle course, is excellent. It exhibits better accuracy for a given number of degrees of freedom than Discontinuous Galerkin approaches to thin shells, and comparable results than subdivision finite elements.

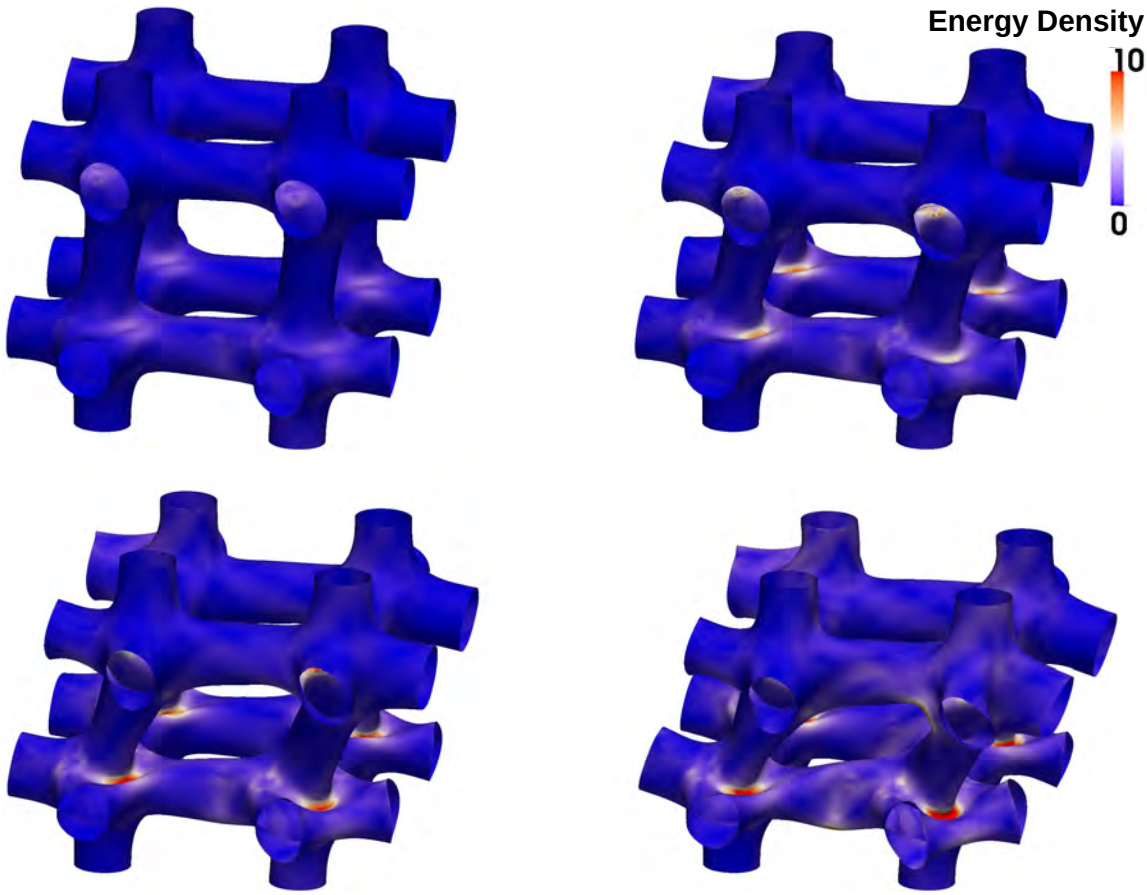


Figure 16: Selected snapshots of the deformation process of the shell with complex topology. Non magnified deformation and colormap of the strain energy density.

2. The proposed method significantly extends the applicability of meshfree methods to thin shell analysis, since existing methods are limited to very simple surfaces admitting a single parametric space. Moreover, it is very robust and general, and can also deal with shells of very complex geometry and topology. This feature has been illustrated by an example of a shell of complex topology.

3. We derived an explicit formula for the second spatial derivatives of LME basis functions, which are defined in terms of nodal locality parameters β_a. We have shown that these basis functions can handle unstructured and non-uniform grids of points.

4. We have derived the full tangent matrix in the geometrically exact theory of Kirchhoff-Love shells, and implemented it in the context of Newton's method.

5. Algorithms to treat surfaces of complex topology and geometry have been implemented in C++. The assembly of stiffness and tangent matrices, the computation of the internal forces and nonlinear iterative solvers have also been coded. Furthermore, an OpenMP parallelization has been performed at the patch level to accelerate the matrix assembly and the internal forces computation. The Math Kernel Library from INTEL has been extensively used to solve the system of equations by direct methods.

Ongoing research

Phase-field modeling of fracture mechanics in linear thin shells

A phase-field approximation for brittle fracture was introduced by Francfort and Marigo (1998) for mathematical purposes, and has gathered significant interest in computational fracture. This approximation reduces considerably the implementation complexity because the numerical tracking of the fracture is not needed, at the expense of a high computational cost. We are currently extending these approaches to deal with brittle fracture in thin shells. In the simplest model, the phase-field is assumed to be constant across that thickness, resulting in the energy functional for a thin shell (elastic and surface energies)

$$\Pi[\boldsymbol{u}, \upsilon] = \int_{\Omega_0} (\upsilon^2 + \eta_{\bar{\kappa}}) \mathcal{W}(\boldsymbol{u}) \; d\Omega_0 + \int_{\Omega_0} G_c \; h \left[\frac{(1-\upsilon)^2}{4\bar{\kappa}} + \bar{\kappa} |\nabla \upsilon|^2 \right] d\Omega_0 + \Pi_{\texttt{ext}}[\boldsymbol{u}],$$

where υ is phase-field describing the cracks, G_c is the critical energy release rate, or surface energy, in Griffith's theory, and $\bar{\kappa}$ is a positive regularization constant to regulate the size of the fracture zone. It was shown by Bourdin et al. (2008) for bulk materials, that when $\bar{\kappa}$ tends to zero, the phase-field approximation of the fracture energy converges to the sharp fracture energy. By minimizing this energy with respect to the mechanical displacements and the phase-field, subject to an irreversibility condition to avoid crack healing, this model can describe crack nucleation, propagation, branching and merging.

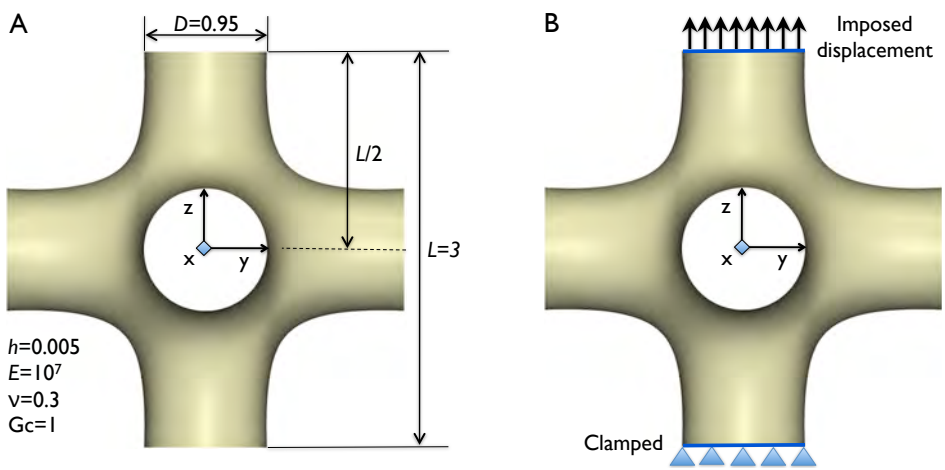


Figure 17: Sketch for the brittle thin shell problem. (A) Material and geometrical parameters describing a set of six connected open pipes. (B) The prescribed boundary conditions and the applied incremental displacement.

We are currently collaborating with Professor Timon Rabczuk and Fatemeh Amiri at Bauhaus University Weimar in Germany. Here, we show a preliminary experiment with a thin shell with complex topology. Figure 17 shows the surface, consisting of a set of six connected open pipes. The boundary curve at the bottom is clamped, whereas the top boundary curve is incrementally displaced in the upward $(0, 0, 1)$ direction. Material parameters have been selected as $E = 10^7$ and $\nu = 0.3$, $G_c = 1$, while the thickness is $h = 0.005$. The discretization of the geometry (geometric markers) and the control points consists of two arrangements of 25668 and 100380 unstructured set of points respectively. The original coarse point-set has been obtained from the MATLAB Central File exchange. The phase field is represented as a colormap on the reference configuration, see

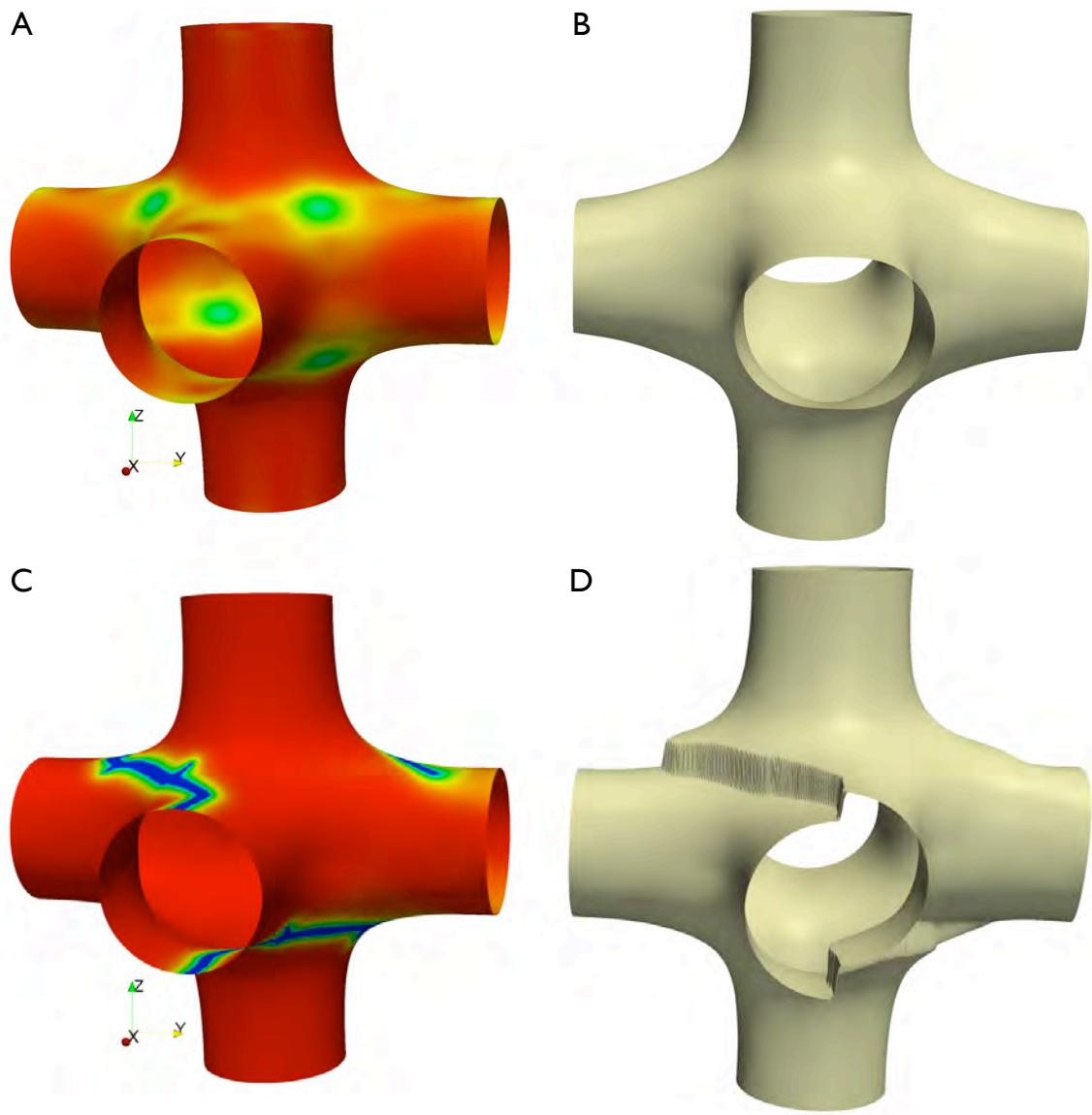


Figure 18: Selected snapshots of the deformation process of a brittle thin shell with complex topology. The boundary curve of the bottom pipe is clamped and the top boundary curve is incrementally displaced in the upward $(0, 0, 1)$ direction. The process has been performed without an initial crack. (A,C) Phase field as colormap in the reference configuration for two selected imposed displacements of the top boundary curve, just before the fracture ($d = 0.0055$) and for the final imposed displacement, $d = 0.01$. (B,D) Deformed configurations for two selected instants, the deformation field has been magnified by 20.

Figs. 18A,C. Figures 18B,D shows the physical deformation obtained, which has been amplified by a factor of 20 to give a better idea of the resulting displacements. Due to the nonlinearity of the model, we observe a symmetry-breaking solution.

Open lines for research

We next discuss some open ideas for research derived from the work performed:

(i) In structural analysis, we commonly face either complex boundary constraints and internal connections (non-manifold surfaces (Green and Turkiyyah, 2004; Cirak and Long, 2011)), or problems where contact is involved. We plan to deal with these issues within the current framework.

(ii) A robust NLDR technique, which preserves the isometry as much as possible, is lacking. This fact is especially notorious with both highly nonuniform and doublly curved sets of points, e.g. the armadillo point-set surface (Turk and Levoy, 1994). We think that our in-house iterative Nonlinear Local Isometric Embedding (NLIE) method, see Appendix C, can face the issue. Further studies to substantiate this claim are needed.

(iii) A high fidelity (exact CAD-geometry) description of domain boundaries by blending B-Splines and LME has been recently proposed (see Rosolen and Arroyo (2012)). We think that this description can be integrated in the proposed methodology to avoid the ghost nodes at the boundary of the thin shells.

(iv) LME approximants only satisfy up to the first order consistency condition. However, the numerical solutions calculated for linear and nonlinear problems involving high order derivatives have shown to be very accurate when the locality parameter is selected within the range $0.6 \leq \gamma \leq 1$ (Millán et al., 2011, 2012). As the accuracy of LME and second order *max-ent* approximants are comparable (Rosolen et al., 2012), suggesting that high order consistency is not the fundamental property endowing *max-ent* approximants with their excellent approximation properties. We believe that there is enough numerical evidence in these problems to motivate a theoretical study.

(v) The proposed framework allows us to address problems involving high order spatial derivatives on surfaces, like the Canham-Helfrich-Evans energy or the interfacial fluid dynamics in lipid bilayers.

(vi) Thin shell problems are naturally ill-conditioned. The lack of good preconditioners reduces the range of applicability of iterative solvers. Moreover, mesh refinement is usually required to obtain accurate results, especially in highly curved surfaces. This leads to extremely demanding calculations in terms of memory usage for direct solvers. We envision to generate adaptive distribution of points on general point-set surfaces by using Centroidal Voronoi Tessellation (CVT (Liu et al., 2009)). This technique allows us to capture local features of the physical solution(Rosolen et al., 2012), and to refine locally accordingly with either the local curvature or along crack paths. By enhancing our methodology with CVT in conjunction with a high fidelity boundary description, we can greatly extend the scope of meshfree thin shell analysis.

5 Model reduction

The goal of model reduction is to process (simulate, control, optimize) a complex system in terms of a much simpler description. The premise is that, with a proper selection of the reduced model, the essential features of the original system can be preserved by a very light and efficient model. Most often, the reduced model is constrained to evolve in an affine subset of the full configuration space, for instance when a large finite element model is reduced by means of a linear combination of modes selected by Proper Orthogonal Decomposition (POD). Such approaches are common in electrical engineering, molecular simulation, or mechanics (Krysl et al., 2001). The term *nonlinear model reduction* generally refers to the reduced modeling of a nonlinear system, yet relying on a linear (affine) reduced description. Nevertheless, it is apparent that many models evolve on essentially nonlinear manifolds, in particular those involving large amplitude rotations. Rotations, prominent in many mechanical systems, are the paradigmatic example of a nonlinear motion inefficiently described by a linear geometric model. For instance, in three dimensions, nine Cartesian components are required to describe an orthogonal matrix, while only three parameters, e.g. Euler angles, suffice to parametrize a rotation.

What we specifically mean for nonlinear model reduction here is that the reduced description of a given system is not affine, but rather a nonlinear manifold, which we locally reconstruct on the basis of simulation data. This challenges the predominant techniques such as POD, which seek reduced affine models. Often, the term "nonlinear model reduction" refers to the reduction of a nonlinear systems by affine subspaces. Here, we focus on a posteriori reduction based on previous simulation data, or snapshots. Therefore, we aim at data-driven reduction methods, rather than mathematical reduction of the equations governing the system at hand a priori.

Here, we explore the application of NLDR techniques to the reduced modeling of mechanical systems, extending the work of Krysl et al. (2001). In this first study, we restrict ourselves to conservative mechanical systems, and to the forward problem. We follow the classical method of snapshots, in which a training set of simulations, representative of the behavior of the system, is used as the basis for the reduced model. Unlike linear approaches such as the POD (a direct application of PCA), here we consider a nonlinear reduced description, and consequently cannot parametrize the reduced model as a linear combination of modes. We proceed in three steps. First, the snapshots are viewed as a high-dimensional ensemble representative of the configurations of the system, the intrinsic dimensionality of the underlying slow manifold is detected, and the snapshots are embedded in low-dimensions by NLDR. Note that a mere low-dimensional embedding of the snapshots is not enough for our purpose, as it does not provide a parametrization of the slow manifold. For this reason, as a second step, we construct a smooth parametrization of the slow manifold (a map from the low-dimensional embedding to high dimensions) with local maximum-entropy approximants (Arroyo and Ortiz, 2006). A number of key features of this approximation method are essential for the generality and success of the proposed method, namely its ability to provide smooth approximations from scattered unstructured data in any spacial dimension, adapted to the output of NLDR, and to filter noisy data robustly. As the third step, the embedding dimensions are then viewed as generalized coordinates of the system, in terms of which we express the dynamics. Although the real system evolves in a high dimensional space and departs from the slow manifold during its dynamics, we constrain the dynamics of the reduced system to the slow manifold.

There have been extensions beyond linear representations of reduced models, *à la* PCA, in computational mechanics. For instance, POD methods (Chinesta et al., 2010) address high-dimensional problems by finding iteratively new basis functions of separated variables, resulting in a nonlinear approximation method. Farhat and co-workers have proposed methods to interpolate parametrized reduced models, which form a nonlinear manifold (Amsallem and Farhat, 2011).

The present Section is organized as follows. First we formulate the Lagrangian mechanics of a system constrained to evolve in a manifold described by a set of generalized coordinates. We exercise

the discrete Lagrangian mechanics formalism to derive variational time-integration schemes for systems with configuration-dependent mass. Then, we develop a proof-of-concept example, highlighting the potential and limitations of reduced order modeling based on NLDR. Finally, we put forward the major contributions and discuss some future lines for research.

Notation

In this section a different notation respect to the preceding ones is considered. The chosen notation is similar to the employed in classical mechanics, which is in concordance with the area of knowledge where commonly model reduction is applied. Next a detailed description between both notations

X	$\leftrightarrow$	P	set of control points in the high-dimensional physical space, $\mathbb{R}^D$,
x_a	$\leftrightarrow$	$\boldsymbol{P}_a$	control point in $\mathbb{R}^D$,
x	$\leftrightarrow$	$\boldsymbol{x}$	point in $\mathbb{R}^D$,
Q	$\leftrightarrow$	Ξ	set of control points in the low-dimensional embedded space, $\mathbb{R}^d$,
q_a	$\leftrightarrow$	$\boldsymbol{\xi}_a$	control point in $\mathbb{R}^d$,
q	$\leftrightarrow$	$\boldsymbol{\xi}$	point in $\mathbb{R}^d$,
$\mathbb{X}$	$\leftrightarrow$	$\boldsymbol{\varphi}$	mapping from the embedded space to the physical space,
V	$\leftrightarrow$	Π	potential energy,
P	$\leftrightarrow$	$\boldsymbol{V}$	eigenvector matrix,
$q(x)$	$\leftrightarrow$	$\boldsymbol{\pi}(\boldsymbol{x})$	closest point projection (see Eq. (3)).

Furthermore, to simplify the present method we restrict ourselves to problems only described by one patch.

5.1 Lagrangian mechanics in generalized coordinates

Constrained dynamics

Let $x \in \mathbb{R}^D$ denote the position in Cartesian coordinates of a mechanical system described by its Lagrangian as follows

$$L(x, \dot{x}) = T(\dot{x}) - V(x) = \frac{1}{2}\dot{x}^T M \dot{x} - V(x),$$

where T is the kinetic energy, V the potential energy and M is the constant mass matrix. The corresponding Euler-Lagrange equations can be written as

$$M\ddot{x} = -\frac{\partial V}{\partial x}(x).$$

We are interested in the constrained dynamics on a configuration manifold $\mathcal{M} \subset \mathbb{R}^D$ of dimension $d \ll D$. Let $q \in \mathscr{A} \subset \mathbb{R}^d$ be the generalized coordinates given by NLDR as described in Section 3. We parametrize $\mathcal{M}$ in terms of these generalized coordinates as $x = \mathbb{X}(q)$. By the chain rule, $\dot{x} = D\mathbb{X}(q)\ \dot{q}$, and therefore we can formulate the constrained dynamics as

$$L(q, \dot{q}) = \frac{1}{2}\dot{q}^T \widetilde{M}(q)\ \dot{q} - \widetilde{V}(q),$$

where

$$\widetilde{M}(q) = D\mathbb{X}(q)^T M\ D\mathbb{X}(q) \qquad \text{and} \qquad \widetilde{V} = V \circ \mathbb{X}. \tag{5}$$

Since the mass matrix is in general configuration-dependent, this is an instance of non-separable Lagrangian. By Hamilton's principle, the Euler-Lagrange equations of the reduced system, expressing

the balance of linear momentum in curvilinear coordinates, are

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = 0.$$

In the special case of PCA (or other linear methods), the manifold parametrization is affine, $D\mathbb{X} = P$, and consequently the reduced mass matrix is not position dependent, $\widetilde{M} = P^T M P$.

Discrete Lagrangian mechanics

We resort to variational principles to integrate in time the Euler–Lagrange equations; full details can be found in Marsden and West (2001) and Lew et al. (2004), and references therein. Starting with the action functional

$$S[q] = \int_{t_0}^{t_1} L(q, \dot{q})\ dt,$$

for the system's trajectory $q(t)$ in $\mathscr{A}$ satisfying the initial conditions, variational integrators are based on Hamilton's principle, which states that the trajectories are stationary points of the action, $\delta S[q, \delta q] = 0$ for all admissible δq.

In time-discrete Lagrangian mechanics (one step methods), the action integral S is replaced by an action sum

$$S_d = \sum_{j=0}^{n-1} L_d(q_j, q_{j+1}),$$

where $L_d : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ is the discrete Lagrangian, a function of two points $q_0 = q(t_0)$, $q_1 = q(t_1) \in \mathscr{A}$ approximating the action in an interval as

$$L_d(q_0, q_1) \approx \int_{t_0}^{t_1} L(q, \dot{q})\ dt. \tag{6}$$

The discrete Euler-Lagrange equations –the time-integration algorithm–, follow from making the action sum stationary, e.g. $\partial S_d / \partial q_k = 0$, and are given by

$$D_1 L_d(q_k, q_{k+1}) + D_2 L_d(q_{k-1}, q_k) = 0. \tag{7}$$

The accuracy of the variational integrator is given by the order of the quadrature in Eq. (6). Specific numerical integration schemes can be obtained from this formalism. For concreteness, we derive here the discrete Euler-Lagrange equations for the generalized midpoint rule with configuration-dependent mass. Additionally, in Appendix C, the resulting expressions for the trapezoidal rule for a configuration-dependent mass are given. For the generalized midpoint rule, the discrete Lagrangian approximating the action integral between q_0 and q_1 is given by

$$\begin{aligned} L_d(q_0, q_1) &= \Delta t\ L\left((1-\alpha)q_0 + \alpha q_1, \frac{q_1 - q_0}{\Delta t}\right) \\ &= \Delta t \left\{ \frac{1}{2}\left(\frac{q_1 - q_0}{\Delta t}\right)^T \widetilde{M}\Big((1-\alpha)q_0 + \alpha q_1\Big)\left(\frac{q_1 - q_0}{\Delta t}\right) - \widetilde{V}\Big((1-\alpha)q_0 + \alpha q_1\Big)\right\}, \end{aligned} \tag{8}$$

where $\alpha \in [0,1]$. From Eqs. (7,8), we obtain the following equation for q_{k+1}

$$\begin{aligned}
\frac{\partial S_d}{\partial q_k} &= \Delta t \left\{ (1-\alpha) D_1 L \left((1-\alpha) q_k + \alpha q_{k+1}, \frac{q_{k+1}-q_k}{\Delta t} \right) - \frac{1}{\Delta t} D_2 L \left((1-\alpha) q_k + \alpha q_{k+1}, \frac{q_{k+1}-q_k}{\Delta t} \right) \right\} \\
&\quad + \Delta t \left\{ \alpha D_1 L \left((1-\alpha) q_{k-1} + \alpha q_k, \frac{q_k - q_{k-1}}{\Delta t} \right) + \frac{1}{\Delta t} D_2 L \left((1-\alpha) q_{k-1} + \alpha q_k, \frac{q_k - q_{k-1}}{\Delta t} \right) \right\} \\
&= \Delta t \, (1-\alpha) \left\{ \frac{1}{2} \left(\frac{q_{k+1}-q_k}{\Delta t} \right)^T D\widetilde{M}\Big((1-\alpha) q_k + \alpha q_{k+1} \Big) \left(\frac{q_{k+1}-q_k}{\Delta t} \right) - D\widetilde{V}\Big((1-\alpha) q_k + \alpha q_{k+1} \Big) \right\} \\
&\quad + \Delta t \, \alpha \left\{ \frac{1}{2} \left(\frac{q_k - q_{k-1}}{\Delta t} \right)^T D\widetilde{M}\Big((1-\alpha) q_{k-1} + \alpha q_k \Big) \left(\frac{q_k - q_{k-1}}{\Delta t} \right) - D\widetilde{V}\Big((1-\alpha) q_{k-1} + \alpha q_k \Big) \right\} \\
&\quad - \widetilde{M}\Big((1-\alpha) q_k + \alpha q_{k+1} \Big) \left(\frac{q_{k+1}-q_k}{\Delta t} \right) + \widetilde{M}\Big((1-\alpha) q_{k-1} + \alpha q_k \Big) \left(\frac{q_k - q_{k-1}}{\Delta t} \right) \\
&= 0.
\end{aligned} \tag{9}$$

These calculations require the directional derivative of the reduced mass matrix, and derivative of the reduced potential energy, computed as follows from the product and the chain rule

$$D\widetilde{M}(q)\ \dot{q} = \mathbb{Y}^T(q,\dot{q})\ M\ D\mathbb{X}(q) + D\mathbb{X}(q)^T M\ \mathbb{Y}(q,\dot{q}),$$

with

$$\mathbb{Y}(q,\dot{q}) = D^2\mathbb{X}(q)\ \dot{q}, \tag{10}$$

and

$$D\widetilde{V}(q) = DV(\mathbb{X}(q))\ D\mathbb{X}(q).$$

It is also possible to obtain an expression for the generalized velocities associated to the Cartesian velocities at a point in $\mathcal{M}$ and tangent to it

$$\dot{q} = \left[D\mathbb{X}(q)^T\ D\mathbb{X}(q) \right]^{-1} D\mathbb{X}(q)^T\ \dot{x}. \tag{11}$$

It is clear from Eq. (10) that this formulation requires a smooth representation of the slow manifold.

From a practical viewpoint, the choice $\alpha = 0$ is most convenient since then Eq. (9) is merely quadratic in q_{k+1}. Otherwise, the resulting algorithm is implicit in q_{k+1} in a very nonlinear way, requiring third order derivatives of $\mathbb{X}(q)$ for an iterative solution with Newton's method. The implementation details for $\alpha = 0$ in combination with Newton's method are given in Appendix C.

5.2 Example: Neo-Hookean hyperelastic solid

We show the capabilities of the present method in a proof-of-concept problem, the dynamics of a hyperelastic neo-Hookean body undergoing large deformations. We compare the proposed nonlinearly reduced dynamics against the standard reduced dynamics from a linear method (PCA). The geometry of the object and mesh are shown in Fig. 19.

We concisely describe here the main steps required to describe the data-driven slow manifold $\mathcal{M}$, namely (1) the computation of a low-dimensional embedding of a training set of snapshots, and (2) the definition of a smooth parametrization of the slow manifold embedded in high dimensions. Before to obtain the low-dimensional embeddings we perform two important tasks, the collection of precomputed offline configurations to build the training set, and the study of the intrinsic dimensionality d of the embedded manifold. Then, we compute the low-dimensional embedding of the training set applying dimensionality reduction methods. Next, we detail the main features involved in the discretization of the parametrization. Finally, we quantify the quality of the reduced dynamics, we compare the real trajectory against the standard linear reduced dynamics from PCA and the nonlinear reduced dynamics from our methodology.

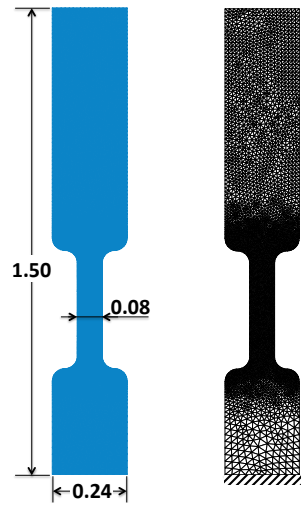


Figure 19: Sketch for the neo-Hookean hyperelastic solid in 2D (left) and the mesh used to compute the elastodynamics (right). The object is clamped at the bottom, and subject to gravity. The material parameters are $\lambda = 1000$ and $\mu = 500$, and the model is discretized by 9428 linear triangular elements and 4934 nodes.

Training set

We assume now that, on the basis of an offline set of calculations, we have a configuration ensemble made out of simulation snapshots. In practice, a decimation step is necessary to define a workable training set, which we denote by $X = \{x_1, x_2, \ldots, x_N\} \subset \mathbb{R}^D$. We define the ensemble such that any two snapshots are at least a cut-off distance away from each other. Unlike PCA, which requires the snapshots to be full vectors in $\mathbb{R}^D$, the method presented here allows us to prune the snapshots into coarser representations before we apply NLDR. For instance, it is easy to simplify systematically a large finite element mesh to a set of landmark points sampling the boundary, edges, or the skeleton, and thus decrease D significantly. This allows us to avoid the manipulation and storage of large amounts of data.

We simulated 35 trajectories for different initial conditions (position and velocity), with an explicit mid-point rule integration scheme ($\Delta t = 2 \cdot 10^{-5}$). For each trajectory, a total number of $4 \cdot 10^6$ time steps were computed and sampled with a frequency of 1/1000, resulting in 4000 snapshots per trajectory. In an offline process, the first 500 snapshots were deleted to avoid poorly sampled regions arising from unlikely starting points. Further, due to the symmetry of the problem, each retained snapshot was duplicated and mirrored. As a result, a total of 245,000 full configurations in $D = 4934 \times 2$ were obtained. These snapshots were then pruned by keeping only the nodes at the boundary, resulting in a simplified snapshots in dimension $D = 868$, approximately one tenth of D. In pruning the snapshots, one needs to balance compression and fidelity. For instance, we found that keeping only 100 points along the skeleton resulted in a poorer description of the dynamics. Finally, we decimated the pruned training set, down to 50,000 configurations in $D = 868$.

Intrinsic dimension

As discussed in Section 2, we can *a priori* estimate the intrinsic dimension underlying the data set. Figure 20 illustrates how the correlation dimension (A) and the cumulative variance error of local PCA (B) provide hints, but not conclusive information for realistic training sets. These method are blind to the origin of the data. The intrinsic dimension appears to be between 2 and 4, depending

on the scale of observation. We evaluate the dimension *a posteriori* (after dimensionality reduction on the parametrization of the slow manifold) by tracking the reconstruction error as a function of the dimension of the embedding, see Fig. 20C. With a parametrization of the slow manifold at hand, it is possible to evaluate its quality by comparing high dimensional data with their reconstruction through the parametrization. Ideally, this is done with a test set $Z = \{z_1, \ldots, z_M\} \subset \mathbb{R}^D$, different and possibly larger than the training set, sampling $\mathcal{M}$. We can then define the relative reconstruction error

$$e = \frac{1}{M} \sum_{a=1}^{M} \frac{|z_a - \mathbb{X}(q(z_a))|}{|z_a|}, \tag{12}$$

where $q(x)$ maps high-dimensional points to low-dimensional representations, see Appendix B. Additionally, with PCA it is straightforward to increase the dimension, while for the nonlinear methods, whose objective is to keep d as small as possible, we have only constructed one- and two-dimensional slow manifolds. As expected for data with nonlinear correlations, it can be observed that the reconstruction error stemming from NLDR are significantly smaller than that given by PCA for a given dimension. The dimensionality estimations are consistent with the *a priori* methods. From our experience, the most practical method of dimensionality estimation is tracking the cumulative variance error of local PCA. It is fast, easily made parallel, can be performed on pruned data, and does not need a low-dimensional embedding of the training set.

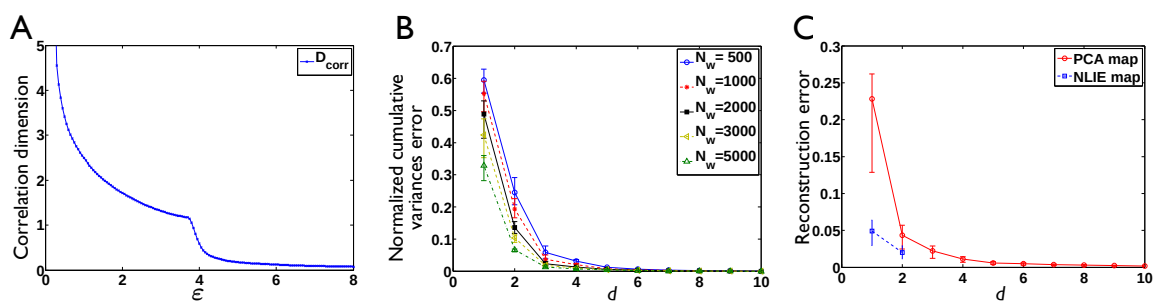


Figure 20: (A) The correlation dimension method considers a closed ball of radius ε at the center of each data point, and counts the number of points inside of this ball. The dimension is estimated by noting that the average number of counts $C(\varepsilon)$ should grow linearly with ε for 1D objects, quadratically for 2D entities, and so on, i.e. $C(\varepsilon) \propto \varepsilon^d$. It is a scale-dependent quantity that should be interpreted carefully a: at very large ε, $d_{corr} \approx 0$ as the data set looks like a single point, while for small ε, the dimension is overestimated by the presence of noise. (B) Mean of the normalized cumulative variance error from local PCA ($\pm$ 25th and 75th percentiles), $\left(1 - \sum_{i=1}^{d} \lambda_i\right) / \sum_{i=1}^{D} \lambda_i$, computed in small local subsets of the data where N_w is the window size. λ_i denote the eigenvalues of the local covariance matrix arranged in decreasing order. The intrinsic dimension d is selected such that it preserves a given fraction of the variance of the original data. In contrast with the global correlation dimension, this method provides a local estimation of the intrinsic dimension. (C) Mean of the reconstruction error as defined in Eqs. (1,12) ($\pm$ 25th and 75th percentiles). The reconstruction error is a measure of the dissimilarity between the original high-dimensional data points and their reconstruction through the slow manifold parametrization.

Dimensionality reduction

Figure 21 shows the two- and three-dimensional embeddings of the training set by PCA, Isomap and NLIE. It is obvious from the three-dimensional embeddings that the data has highly nonlinear correlations, i.e. the slow manifold is highly curved. As a result, it is not surprising that Isomap collapses some parts of the training set. In contrast, NLIE gently unrolls the curved manifold by

introducing local distortions that minimize the global stress energy. A close visual inspection of the three-dimensional embedding provided by PCA or NLIE suggests that the data lies on a "thick" elongated hyper-surface.

The one-dimensional embedding of PCA can be found by projecting along q_1 the two-dimensional embedding. It is clear from the figure that this method will collapse far-away conformations, destroying the geometric structure of the slow manifold. In contrast, the one-dimensional NLIE embedding provides a coarse, yet meaningful representation of the model.

We finally note that, in principle, it is possible to define a nonlinear parametrization of the slow manifold such as that in Eq. (2) on the basis of PCA embeddings. However, from our experience in this and other applications, these low-dimensional embeddings introduce larger distortions and are not robust for significantly curved training sets.

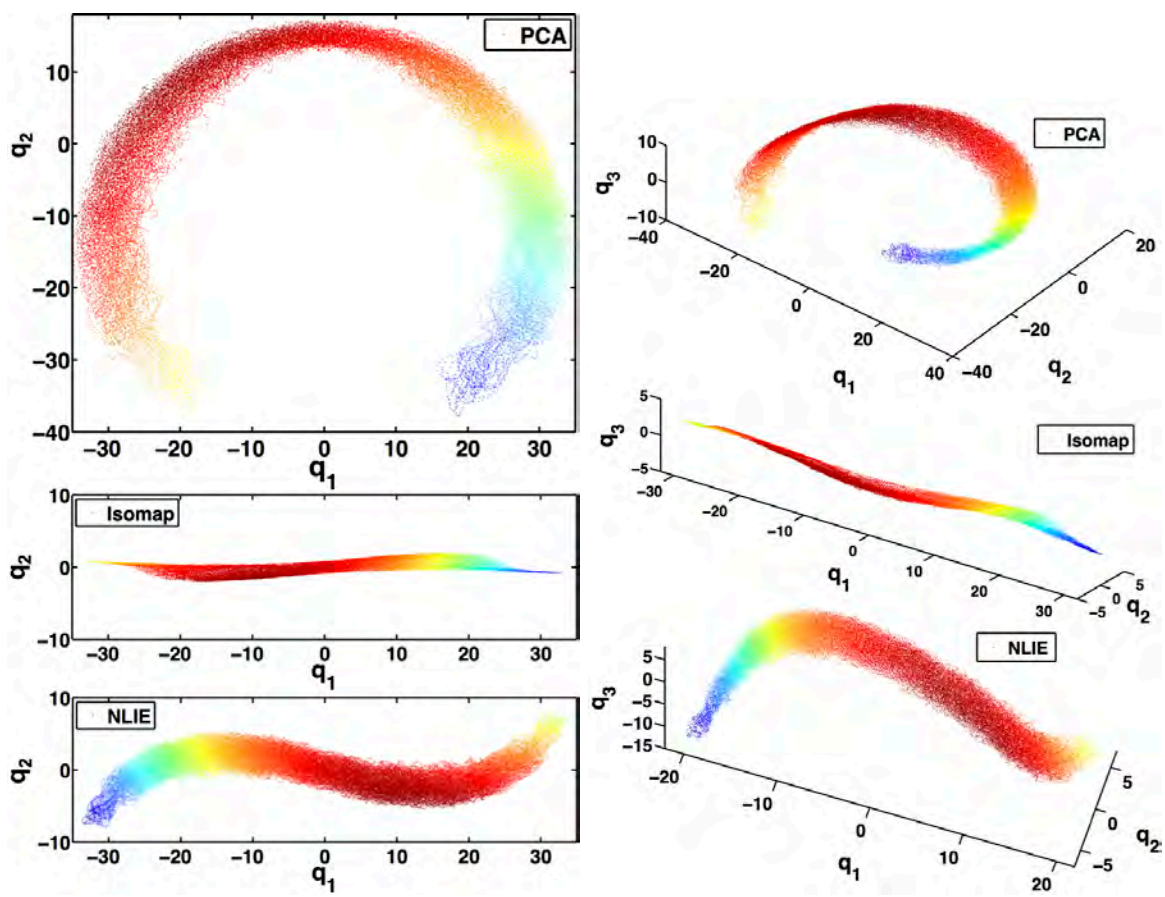


Figure 21: Two- and three-dimensional embeddings obtained with PCA (top), Isomap (center) and NLIE (bottom). The colormap, indicating the Euclidean distance to the first configuration of the training set, is only provided as visual guide.

Parametrization of the slow manifold

We follow Eq. (2) to parametrize smoothly the slow manifold from the low-dimensional embedding, $\mathbb{X}(q)$, with LME basis functions. Details about the calculation of these meshfree approximants and their spatial derivatives can be found in the works by Arroyo and Ortiz (2006); Rosolen et al. (2010).

The noise filtering can be controlled with the width of the LME basis functions. Figure 22 shows in more detail the two-dimensional embedding found with NLIE, where the red points represent the set Q. An obvious drawback of defining $\mathbb{X}(q) = \sum_{a=1}^{N} p_a(q) x_a$ is that N can be very large, here 50,000, and therefore the evaluation of this mapping, required at each time-step, can be quite expensive. In fact, this mapping parametrizes a relatively simple slow manifold, and the large number of samples is only required to provide a statistically representative low-dimensional embedding. For computational convenience, we subsample the description of the slow manifold as follows. We divide the low-dimensional embedding space in equally spaced bins of size $h_{bin} = L/n_{bin}$, where L is the length of the low-dimensional embedding along q_1 and n_{bin} the number of bins in this direction. h_{bin} should be commensurated to the geometrical features of the slow manifold. This defines a regular grid, from which we retain the points lying in well-sampled regions of the embedding of low-dimension, the blue points in Fig. 22, that we denote by $\hat{Q} = \{\hat{q}_1, \hat{q}_2, \ldots, \hat{q}_{\hat{N}}\}$ with $\hat{N} \ll N$. We then define the local maximum-entropy basis functions associated to $\hat{Q}$, $\hat{p}_b(q)$, with the locality parameter $\beta = 1/h_{bin}^2$ (Arroyo and Ortiz, 2006), and fit the slow manifold with a least-squares criterion (see also Appendix B)

$$\min_{\{\hat{x}_1, \ldots \hat{x}_{\hat{N}}\}} \sum_{a=1}^{N} \left\| x_a - \sum_{b=1}^{\hat{N}} \hat{p}_b(q_a) \hat{x}_b \right\|^2 .$$

In the example shown in Fig. 22, we selected $n_{bin} = 150$, leading to a subsampling of the slow manifold from $N = 50000$ to $\hat{N} = 2115$ control points.

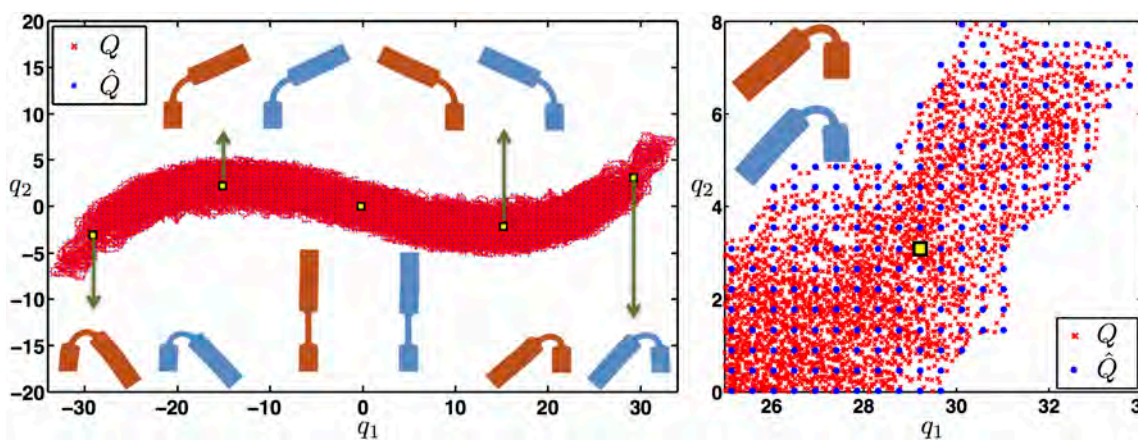


Figure 22: Two-dimensional embedding Q obtained from the original training set X with NLIE (red points), and subsampled node set $\hat{Q}$ (blue points), over the entire two-dimensional embedding (left). For some selected $\hat{q}_b$, we depict the corresponding configuration $\hat{x}_q$ obtained from the least squares fit, and the closest conformation from the original training set. A magnification of the upper-right corner of the two-dimensional embedding (right).

We can time-integrate the reduced dynamics with the methods outlined above and the smooth mapping $\mathbb{X}(q) = \sum_{b=1}^{\hat{N}} \hat{p}_b(q) \hat{x}_b$, leading to a numerical trajectory in the low-dimensional space $q(t)$. The nonlinearly reduced dynamics are fully determined by the effective potential energy $\widetilde{V}(q)$ and the reduced mass $\widetilde{M}(q)$. Figure 23A shows the effective potential, whereas Fig. 23B shows the total reduced mass in the two-dimensional embedding. It can be observed that the reduced mass, which is in fact the square mass-weighted L_2 norm of the derivative of the slow manifold parametrization, see Eq. (5), is nearly constant except at the boundaries of the region of interest. At the boundary, it is not surprising that the quality of the mapping $\mathbb{X}(q)$ is degraded. Yet, these regions are visited

seldom due to their high potential energy.

Figure 23C shows an example of reduced dynamics $q(t)$. Visually, these dynamics retain the qualitative features of the full, high-dimensional dynamics, with high amplitude oscillations. In the examples below, we consider both PCA- and NLIE-reduced dynamics. We always use a midpoint rule with $\alpha = 0$, with a time step $\Delta t = 10^{-4}$ in the reduced simulations (5 times larger than that of the full dynamics). Details about the time-integration algorithm are given in Appendix C.

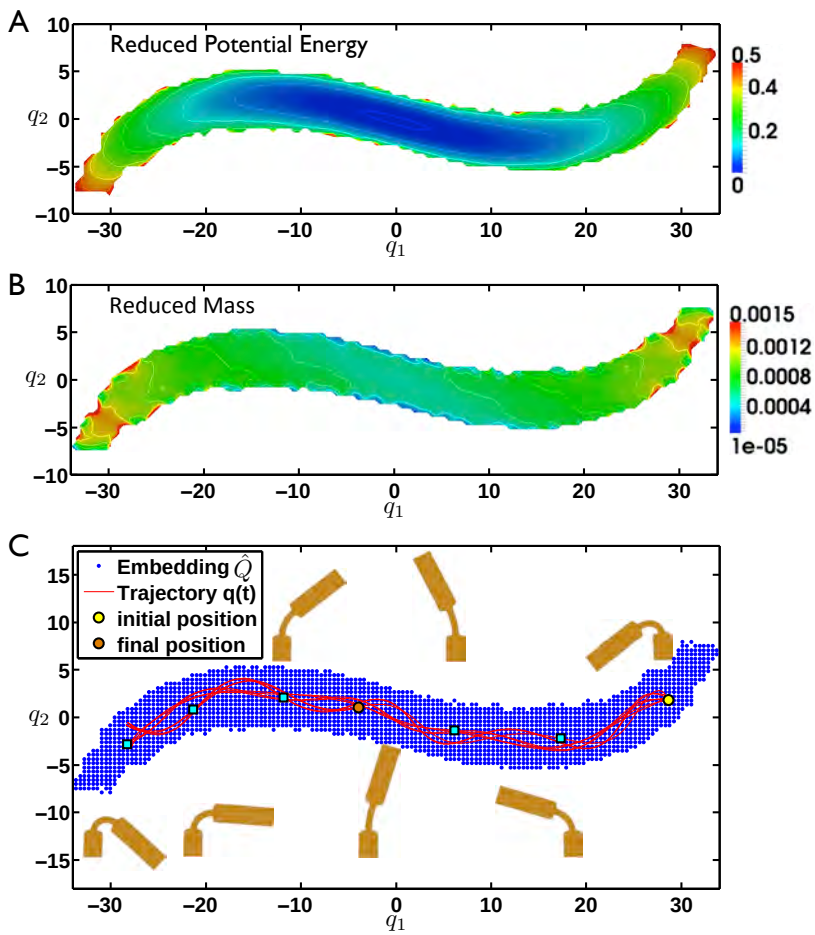


Figure 23: (A) Reduced potential energy $\widetilde{V}(q)$. (B) Total reduced mass $\sum_{i,j} \widetilde{M}_{ij}(q)$. (C) Reduced trajectory in the two-dimensional embedding from NLIE, over seven large amplitude oscillations of the system. The initial, final, and selected snapshot positions are highlighted.

For a more quantitative assessment of the quality of the reduced dynamics, we compare them to the full dynamics, $x(t)$, with the following metric

$$\int_{t_0}^{t_1} D\big(x(t), \mathbb{X}(q(t))\big)\, dt = \int_{t_0}^{t_1} \left(\int_{\Omega_0} \big\| \varphi_h(x(t)) - \varphi_h(\mathbb{X}(q(t))) \big\|^2 dV \right)^{1/2} dt,$$

where here $\varphi_h(x)$ denotes the finite element deformation mapping defined from the nodal positions x. In practice, we sample the time-integral at the time-steps $t_j = j\Delta t$, and compute a non-dimensional

distance between trajectories $\mathcal{D}$ as

$$\mathcal{D} = \frac{\sum_{j=0}^{n} \sqrt{(x_j - \mathbb{X}(q_j))^T M \ (x_j - \mathbb{X}(q_j))}}{\sum_{j=0}^{n} \sqrt{x_j^T M \ x_j}}, \tag{13}$$

where M is the mass matrix of the elastic body computed at the reference configuration.

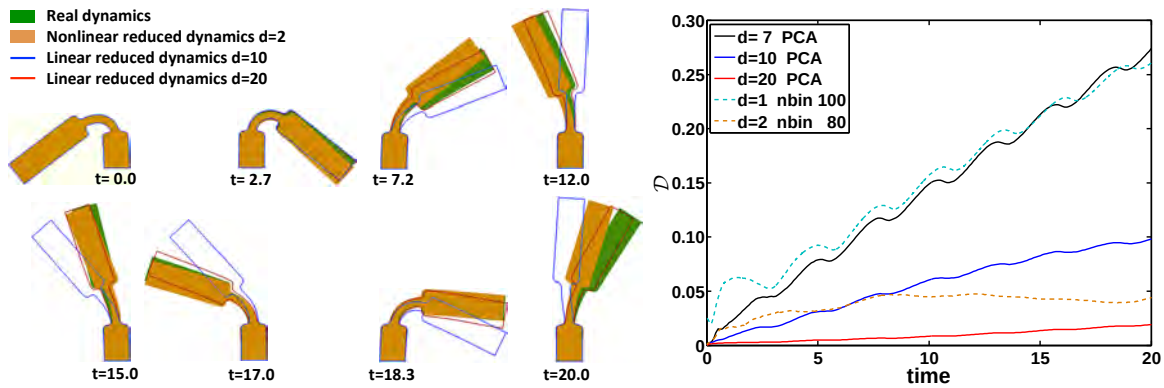


Figure 24: Snapshots during the time evolution of a high-dimensional trajectory $x(t)$, and the reduced trajectories $\mathbb{X}(q(t))$ obtained from PCA and NLIE (left). The quality of the reduced dynamics is measured by the relative cumulative distance, Eq. (13), between $x(t)$ and $\mathbb{X}(q(t))$ (right).

Figure 24 compares, both visually with conformations at selected instants, and quantitatively with $\mathcal{D}$, the full dynamics with the reduced dynamics obtained either by PCA with various dimensions, or by NLIE with dimensions one and two. Opposite to what could be expected from the intrinsic dimensionality analysis, the results indicate that more than 10 PCA modes are needed to capture the dynamics of the system. With only one nonlinear generalized coordinate, we obtain a poor description of the system, comparable to 7 PCA modes. This is not surprising, since the large amplitude oscillations lead to a highly nonlinear configuration manifold, and therefore PCA needs to properly describe its affine hull for reasonable results. As the numbed of PCA modes increases, we obtain accurate dynamics. Strikingly, with only two nonlinear dimensions, we obtain results that lie between those given by 10 and 20 PCA modes. The results suggest that in this example, the affine hull of the slow manifold has tens of dimensions, highlighting the benefit of nonlinear reduced descriptions.

It is interesting that, while correct PCA-reduced dynamics require more than 10 modes, Fig. 20C suggests that, on the basis of the reconstruction error, the slow manifold can be accurately captured by less than 10 modes. This discrepancy highlights the limitations of statistical/geometric quantifications of the dimensionality, which ignore the physics altogether. The section below provides more insight on a related topic.

Energy leak of the slow modes

Figure 25 shows the energy evolution of the full (left) and the reduced (right) dynamics over seven oscillations. It can be observed that, when the system visits its symmetric configuration, the potential energy lies close to its minimum while the kinetic energy is highest. Around this point, high frequency oscillations in the energy can be observed, suggesting a transfer to high frequency modes consistent with the visualization of the dynamics. The reduced dynamics exhibit similar features, but with only one transversal dimension to the main nonlinear coordinate, the features are significantly blunter.

This observation raises the question of the long-time energy behavior of the reduced dynamics.

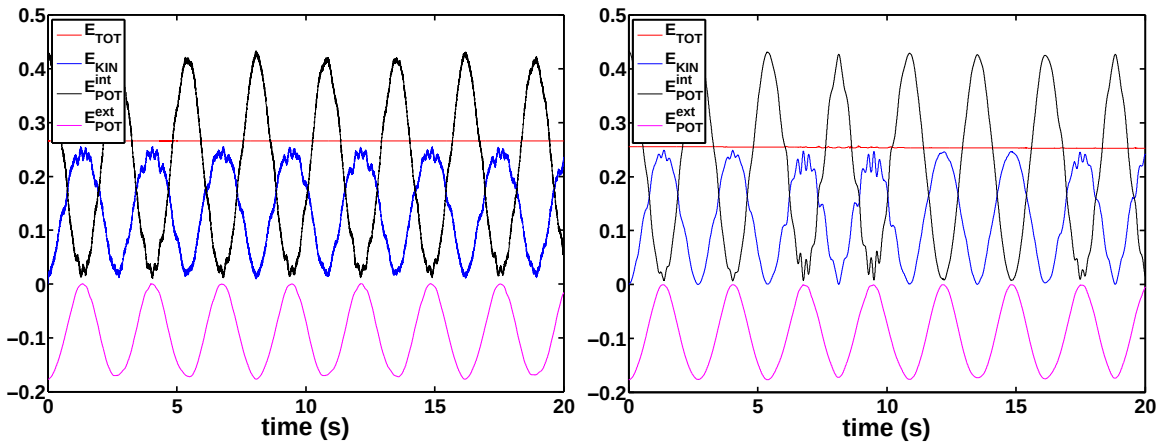


Figure 25: Energy evolution for the full system trajectory (left), and for the reduced trajectory with a two-dimensional nonlinear embedding (right).

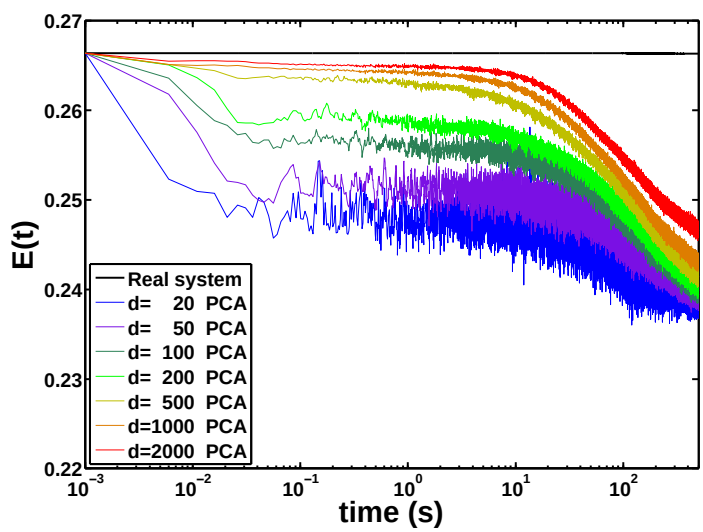


Figure 26: Total energy evolution of the full system and total energy projected onto the reduced models by PCA for different numbers of eigenvectors, $d = 20, 50, 100, 200, 500, 1000, 2000$.

To investigate numerically this issue, we consider a very long trajectory of the full system, and project the dynamics onto the PCA or nonlinearly reduced models. The projection onto the PCA modes is trivial. For the nonlinear model, we use the mapping $q(x)$ described in Eq. (3), together with Eq. (11) to compute $\dot{q}(x, \dot{x})$. We can then compute the full total energy,

$$E(t) = H(x(t), \dot{x}(t)) = V(x(t)) + T(\dot{x}(t)),$$

which is constant up to numerical accuracy since the system is conservative, and the reduced Hamiltonian

$$\widetilde{E}(t) = \widetilde{H}(q(x(t)), \dot{q}(x(t), \dot{x}(t))) = \widetilde{V}(q(x(t))) + \widetilde{T}(q(x(t)), \dot{q}(x(t), \dot{x}(t))).$$

Figure 26 shows the time-evolution of $E(t)$ and $\widetilde{E}(t)$ for $d = 20, 50, 100, 200, 500, 1000, 2000$ PCA modes over a very long simulation. Interestingly, there is a significant and monotonic flow of energy from the low modes into the high-frequency modes. Increasing the reduced dimension, effective at short times, only slightly reduces the energy leak at long times. Once this energy is "thermalized" in the high-frequencies, it does seem to return to the low-frequency modes. Thus, the system described by the reduced coordinates is no longer conservative. These results suggest the need for modeling the net effect of the disregarded degrees of freedom on the reduced dynamics, as is done in the large eddy simulation of turbulence. The long experience in coarse-graining of molecular systems and non-equilibrium statistical mechanics could be useful in this regard (Español, 2003).

Contributions

We summarize here the most significant contributions of this work, to be submitted soon (Millán and Arroyo, 2012):

1. A nonlinear model reduction method that preserves the Lagrangian structure of the original mechanical system has been derived. As a side output, due to the configuration-dependent mass of the nonlinearly reduced model, we have derived appropriate variational integrator schemes.

2. A new iterative NLDR method has been proposed and tested, that we call Nonlinear Locally Isometric Embeding (NLIE). It results from the minimization of a discrete stress energy, which accounts for the differences between the distances computed in the embedded space and in the ambient space. The stress energy is computed taking into account only a moderate number of neighboring points, since we know from geometry that global isometries are not possible in general, leading to frustration and instabilities in methods such as Isomap.

3. We have computed the reduced dynamics in a proof-of-concept test, a nonlinear neo-Hookean material undergoing large deformations in 2D. We confront the real trajectory with the reduced trajectories from a linear method, here PCA, and from a nonlinearly reduced model.

   (a) From the experiment we observe that for PCA, the number of eigenvectors should be at least 10, whereas with 20 the overall system's behavior is very accurate. This large number of eigenvectors contrasts with the estimations of the intrinsic dimension of the ensemble of snapshots, which suggests that the dynamics lie on a slow manifold of dimension between 2 and 4.

   (b) We have shown that few nonlinear collective variables describe the main features of the real dynamics, here $d = 2$. Due to the compact representation, we can rationalize the configuration dependency of the potential energy and the reduced mass, and also describe the trajectory in the reduced configurational space.

4. It is a well known fact that during the dynamics of a nonlinear material, there is significant energy transfer between modes of low and high frequency. We have explored such phenomenon by projecting a real trajectory in the affine spaces provided by PCA. This study brings numerical evidence for further analysis, as the mere increasing of the number of eigenvectors does not seems to compensate the energy leak to high unresolved frequencies. Our study suggests that, analogously to what is done in turbulence, the energy flow to the unresolved scales could be modeled.

Open lines for research

We next discuss some open ideas for research derived from the work performed:

(i) Defining the appropriate metric to measure similarity in high-dimensions determines the quality of the results. From our experience, we feel that, information about the physics of a state should complement geometric information. For instance, the metric we have used in this section cannot tell the difference between the undeformed configuration of the body, and an actual configuration as the system crosses the symmetry point with high kinetic energy and some high-frequency deformation modes. By incorporating, for instance, the potential energy as an extra coordinate, we expect that the method will distinguish configurations close in shape space but with very different potential energy. The output from NLDR should be similar to the embeddings of low-dimension computed previously, but wider. We could define the distance between two configurations as

$$D_{ab}\sqrt{(x_a - x_b)^T M\ (x_a - x_b) + \alpha[V(x_a) - V(x_b)]^2},$$

where the parameter α makes the equation dimensionally consistent, and allows us to weight geometric and potential similarities. From an implementation point of view, this modification implies adding an extra column in training set matrix.

(ii) While nonlinear descriptions are dimensionally more efficient in dealing with the geometric structure of the slow manifold, their dimensionality cannot be easily increased to reduce the representation error. In contrast, PCA can systematically improve by retaining more eigenvectors. We believe that both methods can be reconciled, by defining a PCA-vector bundle on the nonlinear slow manifold. At each neighborhood of the nonlinear manifold, the sharper features/faster modes relevant to that macroscopic state could be identified by local PCA, and used to enrich the description of the data. To create a differentiable structure on the manifold, we could resort to the methods pointed out by Amsallem et al. (2009), and interpolate between the local PCA vector spaces parametrized by the nonlinear slow variables. Such a method could find applications in various fields.

(iii) As suggested by our study of the energy decay of a conservative system projected on a reduced model, we plan to study this energy behavior in more detail. The goal would be to model the effect of the disregarded degrees of freedom on the reduced dynamics through damping and maybe memory terms. We also plan to investigate systems with damping, through D'Alambert principle, which may be more robust to reduction.

6 Quantitative analysis of the Euglenoid movement

The euglenoid movement, or metaboly, is executed by some species of euglenids, with a typical size of 50–100μm (Leander, 2008). It is a peculiar microorganism's motility mode characterized by elegantly concerted, large amplitude distortions of the entire cell with frequencies of about 0.1 Hz. Fundamental questions remain open regarding this movement, including a precise understanding of the actuation mechanism leading to the body shape changes, or even its purpose and efficiency as a swimming strategy. Because all euglenids move primarily with their flagella, in observations of motile cells undergoing metaboly it is difficult to discern the role of the body distortions.

Euglenids are enclosed by a striated surface with two opposing poles called pellicle (see Fig. 27). The ability of euglenids to undergo metaboly has been correlated with the morphology of the pellicle. Flexible pellicles possess a large number of articulated strips (a few tens) often wrapped helically around the cell, whereas rigid pellicles can have as few as four fused strips. Some euglenids exploit their body flexibility to merely round up, elongate, or gently bend. Here we focus on the most extreme oscillatory motions displayed by some species. Detailed experimental observations have shown that shape changes are accompanied by sliding between adjacent pellicle strips (Suzaki and Williamson, 1985, 1986). According to these observations, as the shape changes, the pellicle strips retain their length and width, but change their helicity.

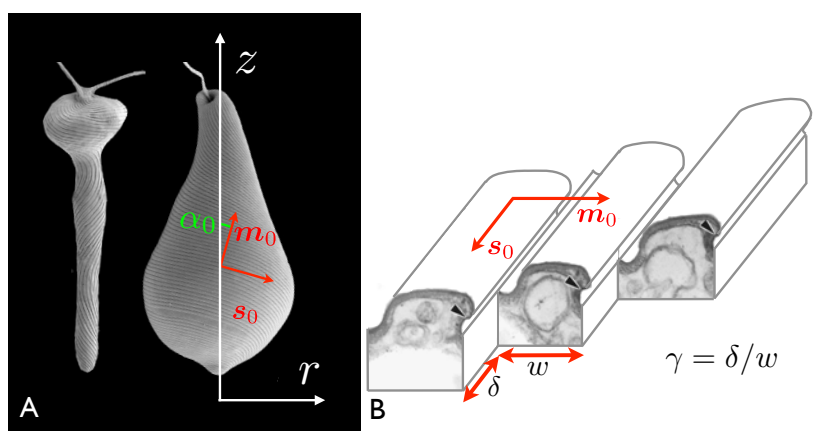


Figure 27: (A) Scanning electron micrographs of the pellicle surface of two flexible euglenids (from Leander (2008)), where the axis of symmetry has been highlighted. The tangent vector fields $\boldsymbol{s}_0$ and $\boldsymbol{m}_0$, along the strips and perpendicular to them, are also illustrated. (B) Ultrastructure of the pellicle (transmission electron micrograph from Leander et al. (2001)), and depiction of the sliding between adjacent strips causing the pellicle shear γ. Further details can be found in Appendix D.

As a first step, our goal here is to understand the biophysical nature of the euglenoid movement and the structure-function relationship of the pellicle. For this purpose, (1) we perform quantitative observations of euglenids executing metaboly caught on video recordings, (2) we then examine these observations in the light of a mathematical model and of numerical simulations of the kinematics, (3) and finally study the hydrodynamics of the euglenoid movement. In this thesis we focus in the first item, whereas the last two are described in Appendix D.

6.1 Quantitative observations

We start from video recordings of four photosynthetic specimens, labelled Euglenid #1 through #4. We assume that these Euglenids remain essentially axisymmetric as they undergo metaboly, which

agrees well with the visual inspection of the videos. A careful checking of movies of Euglenids undergoing metaboly reveals that, while changing shape, the cell body twists non-uniformly around the symmetry axis. Additionally, the movies suggest that the axis of symmetry of the swimmer remains roughly parallel to the focal plane.

From the view point of low Reynolds number (Re) locomotion (Purcell, 1977), a swimming stroke is a closed path in shape space, which needs to be non-reciprocal to accomplish net motion, Fig. 28. In our method, see Fig. 29, we characterize the shape of an axisymmetric swimmer as the region enclosed by its generating curve in a symmetry plane. First, we find the collection of shapes adopted by each swimmer by segmenting and aligning the video frames, and finally we represent numerically the shape of each frame by fitting a B-Spline curve (see Fig. 29A). Then, we identify the geometric structure underlying the set of shapes with Isomap (Tenenbaum et al., 2000). We embed the shapes in 2D and 3D (see Figs. 29B and 30) to examine the nature of the path described by the swimmers in shape space.

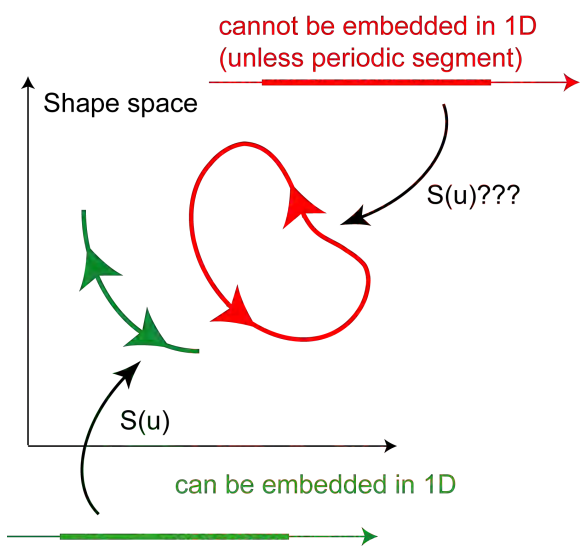


Figure 28: The time-reversibility of the Stokes equations (Re<<1) rules out reciprocal strokes (Purcell's scallop theorem), in which the swimmer follows a given path in shape space in opposite directions alternatively. Without inertia, such a swimmer would stay still after one full stroke (bottom-green). To accomplish net motion the swimming stroke should be closed non-reciprocal path in shape space (top-red).

Video processing

The movies processed here include frames slightly out of focus, are non-uniformly illuminated, and have featured backgrounds. Consequently, our segmentation strongly relies on manual operations. We first enhance the edges of the cell by blurring the surroundings with Adobe Photoshop. This procedure also detaches the flagella from the cell body, and allows us to complete small body parts lying outside of the field of view. We then segment the resulting images with Matlab and ImageJ, and keep the best result. All other computations in the paper are carried out with in-house Matlab codes. The segmented frames are aligned (translated and rotated) with Principal Component Analysis (PCA). The boundary pixels are tagged, and split in two by the symmetry axis detected by PCA. One of the halves is reflected to yield a larger set of points sampling the axisymmetric surface. We

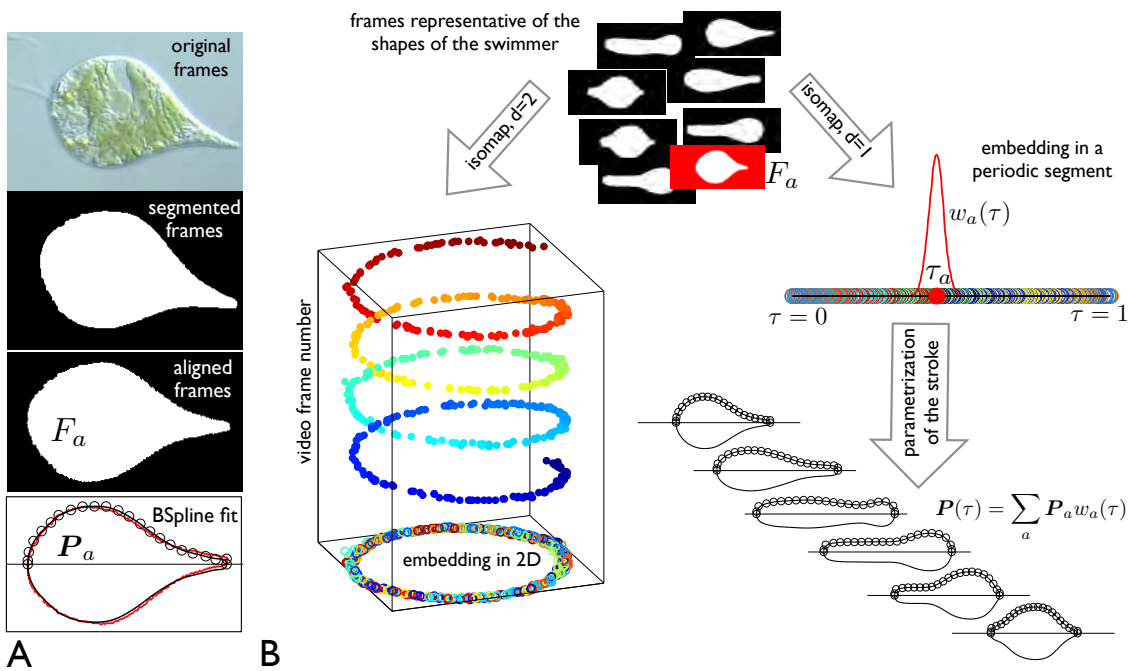


Figure 29: Method for the quantitative analysis of video recordings. (A) The video frames are segmented and aligned to obtain images F_a containing information about the shape alone, devoid of translation, rotation, or textures produced by internal organelles. A B-Spline curve $\boldsymbol{c}_a(u) = \sum_{I=1}^{N} B_I(u)\{r_I, z_I\}_a$ with $u \in [0,1]$, characterized by its control polygon $\boldsymbol{P}_a = \{r_1, z_1, r_2, z_2, \dots r_N, z_N\}_a$ and the B-Spline basis function $B_I(u)$, is fitted to the boundary of F_a, and is a generating curve of the axisymmetric representation of the swimmer. (B) The segmented and aligned frames, representative of the shapes adopted by the swimmer, are embedded in low dimensions by a nonlinear dimensionality reduction technique (here, Isomap (Tenenbaum et al., 2000)). The embedding in 2D shows that the stroke is a closed non-reciprocal path in shape space. Each frame is mapped into a point in the low-dimensional embedding. The colors label the chronological movie frame number. The ability of the algorithm to identify similar shapes from different strokes and define a single geometric stroke is shown by plotting the 2D embedding against the frame number, here for a movie capturing about 4 strokes. The embedding in a periodic 1D segment allows us to smoothly parametrize the stroke in space and pseudo-time as $\boldsymbol{c}(u,\tau) = \{r(u,\tau), z(u,\tau)\} = \sum_a w_a(\tau)\boldsymbol{c}_a(u)$ with $\tau \in [0,1)$, where $w_a(\tau)$ are smooth meshfree basis functions (Arroyo and Ortiz, 2006). At a given τ, the curve $\boldsymbol{c}(\cdot,\tau)$ is a weighted average of the curves fitting video frames whose 1D embedding is in the vicinity of τ.

then fit to these points a B-Spline curve (Piegl and Tiller, 1997) for each frame, expressed as

$$\boldsymbol{c}_a(u) = \sum_{I=1}^{N} B_I(u)\{r_I, z_I\}_a,$$

where $B_I(u)$ are the B-Spline basis functions referred to a reference interval $u \in [0,1]$ and $\{r_I, z_I\}_a$ are the control points associated with the $a-$th frame. We use cubic B-Splines with tangents along the r direction at the end points. The geometry can be adequately described with 20 knot intervals ($N = 23$ control points). However, for accurate resolution of the fluid flow, we consider 30 knot intervals (see Appendix D). The video analysis is carefully checked frame by frame by visual inspection, and the procedure is iterated if needed. Movies S1 and S2 illustrate that this method provides a low-resolution, yet quantitative representation of the strokes.

Parameterization of the stroke

We find low dimensional embeddings of the frames with Isomap (Tenenbaum et al., 2000), using 30 neighbors to approximate geodesic distances. The results are quite insensitive to this parameter in the range 20 to 50. Figure 29B shows how this algorithm identifies similar shapes from up to four different cycles caught in a movie and defines from all the data a single geometric stroke. This algorithm embeds each frame into a point τ_a in a periodic segment scaled to $[0,1]$. This segment represents a geometric pseudo-time. We use this embedding to smoothly parametrize in pseudo-time the generating curve of the swimmer surface during the stroke as

$$\boldsymbol{c}(u,\tau) = \sum_{a=1}^{M} w_a(\tau)\boldsymbol{c}_a(u) = \sum_{I=1}^{N} B_I(u) \sum_{a=1}^{M} w_a(\tau)\{r_I, z_I\}_a,$$

where M is the number of frames, and $w_a(\tau)$ are smooth meshfree maximum-entropy basis functions (Arroyo and Ortiz, 2006). See Fig. 30 for a discussion on the selection of their support size. Figure 31B shows the two-dimensional embeddings of the four swimmers, where each frame is mapped to a point $\boldsymbol{\xi}_a \in \mathbb{R}^2$, together with the reconstruction of the stroke path given by

$$\boldsymbol{\xi}(\tau) = \sum_{a=1}^{M} w_a(\tau)\boldsymbol{\xi}_a.$$

Because our method is based on geometric similarity, τ measures arc-length in shape space rather than physical time. While the displacement of a swimmer after one stroke is invariant with respect to re-parametrizations of time, the hydrodynamic efficiency is not. To estimate the relation between τ and physical time, we use the fact that the movies are shot at constant frame-rate Δt. For each frame associated to the time instant $(a-1)\Delta t$, we find the corresponding pseudo-instant by minimizing $|\boldsymbol{\xi}(\tau) - \boldsymbol{\xi}_a|^2$ with respect to τ. We then identify several distinct strokes and plot the resulting pairs of pseudo-time and physical time, normalized in the interval $[0,1]$ (see Fig. 31E). We fit the one-to-one relation $t = g(\tau)$ with a spline shown in black, and reparametrize the reconstructed stroke in time as $\tilde{\boldsymbol{c}}(u,t) = \boldsymbol{c}\left(u, g^{-1}(t)\right)$, which in an abuse of notation we simply write as $\boldsymbol{c}(u,t)$.

6.2 Quantitative analysis of the shape space

The results from the NLDR show, over the many cycles captured in the movies, well-defined non-reciprocal paths in shape space, the characteristic signature for swimmers at low Re (Fig. 31B). The Euglenids execute the stroke multiple times revealing a well-defined, almost linear relationship be-

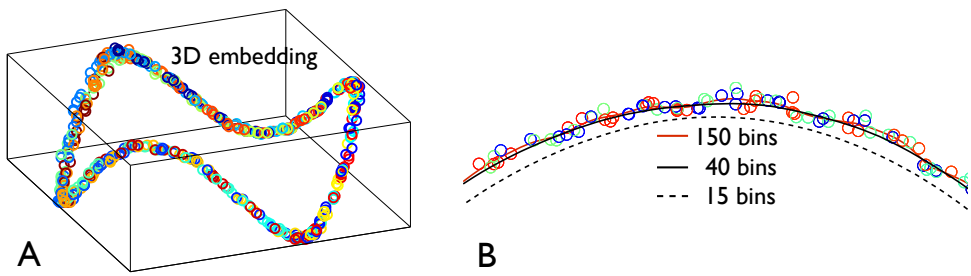


Figure 30: (A) Embedding of the frames in 3D for Euglenid #3, showing that the swimmer indeed follows a closed curve in shape space, and that the 2D picture is not the projection of a more complex trajectory (as a tube). (B) Effect of the width of the basis functions on the reconstructed stroke, here visualized on the 2D embedding. The maximum entropy basis functions (Arroyo and Ortiz, 2006) used here satisfy $w_a(\tau) \propto \exp\left[-(n\tau)^2\right]$, where n is the number of bins used to subdivide the interval $[0, 1)$ and the typical support size of the basis functions is $\sim 1/n$. If n is too large, the reconstructed stroke follows spurious details caused by errors in the data acquisition, whereas if n is too small, then the stroke is excessively smoothed out and shrunk, resulting in less sharply reconstructed swimmers. We find an optimal compromise with $n = 40$.

tween τ and t (Fig. 31E). We reparametrize time accordingly from this point on. With the numerical stroke at hand, we first examine elementary features such as the surface area $S(t)$ and the volume $V(t)$ of the swimmer. We express volume in non dimensional terms of the reduced volume

$$v = 6\sqrt{\pi}V/S^{3/2},$$

the ratio between the cell volume and the volume of a sphere with the same surface area. Thus, v ranges between 0 to 1. We find that the surface area of the four Euglenids remains nearly constant along the stroke, with deviations below 5% (Fig. 31D). This observation is consistent with the hypothesis that pellicle deformation is mediated by sliding of pellicle strips, which retain their length and width (Suzaki and Williamson, 1985). Such a deformation is called simple shear in continuum mechanics, and preserves area locally. The area deviations may be due to the systematic errors in the data acquisition mentioned earlier, or to a slight area deformability of the pellicle. Strikingly, the body volume behaves very differently for Euglenids #1 and #2 and for Euglenids #3 and #4, see Fig. 31B. It is noteworthy that the former exhibit reduced volume deviations of around 20% and average reduced volumes of around 0.75, while the latter show essentially constant reduced volume of smaller magnitude. The volume changing Euglenids necessarily flow significant amounts of fluid in and out of their body quite rapidly during the stroke. Such a volume variation cannot be accommodated by small anatomical features of Euglenids. The motion of the internal organelles during metaboly, clearly seen in the movies, suggests that the volume variations are accommodated by a distributed fluid permeation rather than by localized jets, presumably during the sliding of the strips and through their joints.

Contributions

We summarize here the most significant contributions of this interdisciplinary work, which has been submitted (Arroyo et al., 2012):

1. The methodology developed in this thesis has made it possible to define, from high-dimensional scattered data (here frames in $D \sim 50000$), a smooth one-dimensional parametrization capturing the intrinsic properties of the swimming stroke of Euglenids. Based on this parametrization,

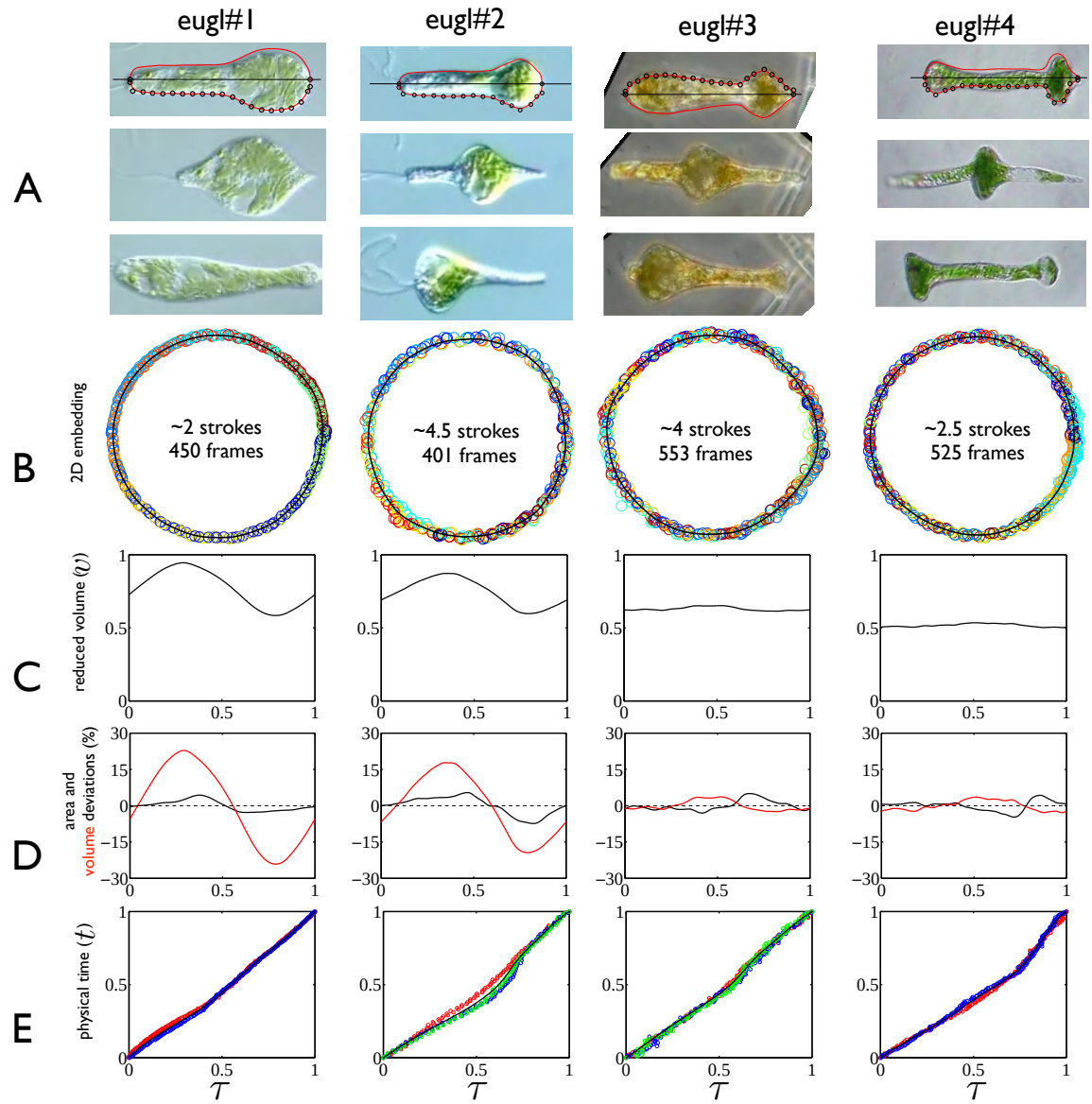


Figure 31: Quantitative analysis of four video recordings: results. (A) Three representative frames for each swimmer, with an illustration of the B-Spline fit to the generating curve of the axisymmetric body. (B) 2D embedding of the frames, where the points are colored (from dark blue to red) chronologically from the video recording. The solid black line shows the smooth reconstruction of the stroke path in this embedding, defined as $\xi(\tau) = \sum_a \xi_a w_a(\tau)$, where ξ_a is the embedding in 2D of frame F_a. (C) Reduced volume v along the reconstructed stroke and its average $\bar{v}$. (D) Area and reduced volume deviations relative to the stroke average along the stroke. (E) Relationship between the scaled physical time (t) and arc-length in shape space (pseudo-time τ) during two (Euglenids #1 and 4) or three (Euglenids #2 and 3) strokes. Data from different strokes are represented in different colors. The spline fit to the data is shown in black.

we have performed a quantitative analysis of metaboly, obtaining a precise description of fundamental biophysical features of this swimming strategy.

Open lines for research

We next discuss some open ideas for research derived from the work performed:

(i) A least squares fitting can be applied to have a lighter representation of the slow manifold, similarly to Appendices B and C.

(ii) It is possible to use the NLDR methods as a classification tool, e.g. the frames for the four microorganisms can be collected to generate a unique embedding. This can shed light on the intrinsic relationships of the shapes hidden in the swimming strokes.

(iii) With higher quality videos, the bottle neck for a fully automated segmentation, a larger number of microorganisms could be considered. The analysis of a large data set could provide new insights into the variability of the stroke within a species, or across the family of Euglenids.

(iv) The proposed method can find many applications in the field of animal and cell motility.

7 Describing molecular conformations with data-driven nonlinear collective variables

One of the central questions in molecular biophysics is the study of the transitions between conformations of large macromolecules, such as proteins. In this area, massive amounts of data are generated from molecular simulations. Extracting understanding from the high-dimensional data from Molecular Dynamics (MD) simulations is a pressing challenge. Visual inspection of the conformation ensemble is a common tool of exploration, but in complex systems presents many limitations and does not bring much insight. An even more fundamental limitation of MD simulations of complex systems is gathering sufficient sampling to be statistically relevant, i.e. to make a connection with thermodynamics. This issue can be illustrated by noting that the typical time-scale of conformational changes ranges from micro to milliseconds and up (rare events), whereas the time-scale of typical bond vibrations, which needs to be resolved, is femtoseconds. This makes an overwhelming 12 orders of magnitude gap. From a statistical mechanics viewpoint, conformations with relatively high energies such as those at saddle points between stable conformations, have an exponentially small probability of being visited in a MD run, leading to poor sampling.

Conformational changes of macromolecules often are described in terms of the so-called collective variables (CV), which in simple molecules can be a single coordinate or a dihedral angle. Collective variables are important to rationalize the behavior of macromolecules, as well as to drive accelerated sampling methods. Collective variables give insight and help in understanding the mechanisms by which a molecular system explores phase space. The computation of the free energy landscape of a molecular system, a fundamental link between MD and thermodynamic observables, is parametrized by CVs, which are instrumental in enhanced sampling methods. There exist a myriad of such techniques; a non-exhaustive list of important methods is Umbrella Sampling and the Weighted Histogram Analysis Method (Patey and Valleau, 1975; Kumar et al., 1992), the Adaptive Biasing Force method (ABF, Darve et al. (2008)), Metadynamics (Laio and Parrinello, 2002), and non-equilibrium work methods (Jarzynski, 1997). Unfortunately, the success of these methods hinges upon a good choice of CV, far from obvious in large macromolecules.

Collective variables are typically chosen out of experience or intuition. Unfortunately, complex macromolecules present highly collective motions, in which physical intuition may not provide adequate guesses for the CVs. Good CVs should (1) provide physical insight, (2) capture the metastability of the system so that they really enhance sampling, and (3) be as few as possible. While the last two items may seem contradictory on the basis of intuition and methods based on PCA, there is evidence that the intrinsic dimensionality of conformational dynamics of macromolecules is rather low (Hegger et al., 2007). This is as consequence of many soft bonds between the atoms at timescales of nanoseconds, which stabilize the molecule and constrain the admissible configurations, reducing dramatically the number of effective degrees of freedom. Due to steric interactions, the CV are expected to be highly nonlinear.

7.1 Systematic determination of CVs

Identifying meaningful CVs in an automatic and robust way has been attempted by many researchers, but is not an easy task. State-of-the art methods rely on normal modes (Tama and Sanejouand, 2001), Path Collective Variables (PCV, Branduardi et al. (2007)) and Principal Component Analysis (PCA). PCA is a standard method in post-processing (Li and Dowell, 2006; Meyer et al., 2006; Maisuradze et al., 2009), and also to enhance sampling (Spiwok et al., 2007). It is important to note that PCA cannot produce compact descriptions of complex macromolecules, and in practice, hundreds of PCA modes are required for accuracy. This can be understood by noting that conformational changes often involve large rotations of one part of the molecule with respect to the other about a molecular

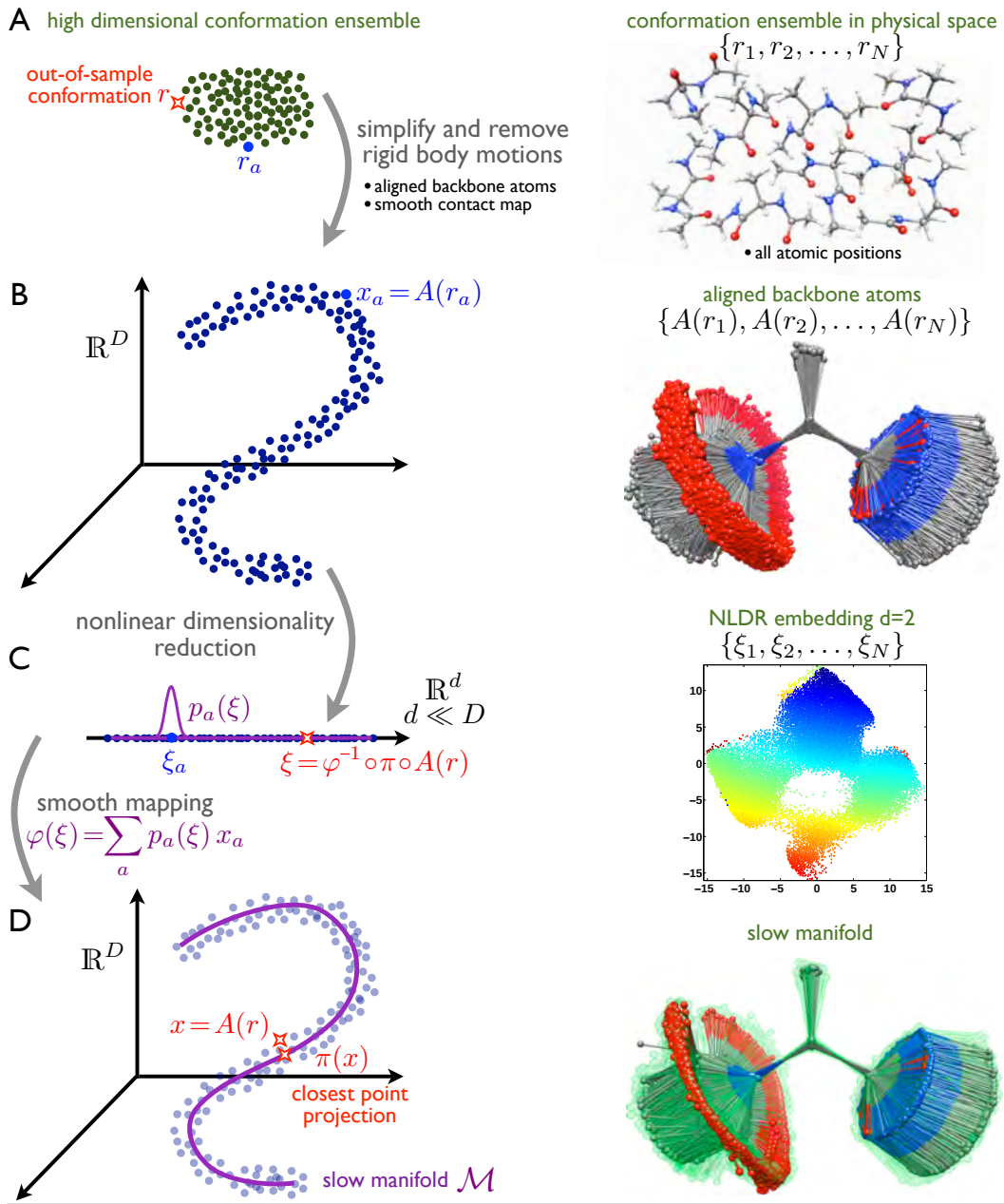


Figure 32: Sketch of the slow manifold parametrization (left), and an illustration of this procedure for the alanine dipeptide (right). (A) High dimensional ensemble built with conformations that populate as much as possible the configurational space. (B) Selected conformations representative of the molecule variability are simplified and the rigid body motions are removed. (C) The conformations are embedded in low dimension by a NLDR technique. (D) With the low-dimensional embedding at hand, the slow manifold hidden in the conformations can be reveled through a smooth parametrization $\varphi(\xi) \in \mathbb{R}^D$. Also, it is possible to define the embedded sample position $\xi(r) \in \mathbb{R}^d$ of the closest point projection $\pi(A(r))$ for a new out-of-sample conformation $r \in \mathbb{R}^D$, where the value $\xi(r)$ is found by solving a nonlinear minimization problem.

hinge. The approximation of a rotation by means of an affine subspace is not the most compact representation, which makes it impossible to accurately and efficiently parametrize the CV along the transition states by PCA. Identifying such CVs and computing some quantities of interest along these, such as the *free energy*, is an important contemporary topic in molecular simulation.

Recently, new developments have emerged, which consider nonlinear CVs either to improve the visualization and data analysis of the massive output (Das et al., 2006; Brown et al., 2008; Ferguson et al., 2010; Ceriotti et al., 2011), or to accelerate molecular dynamics simulations (Spiwok and Králová, 2011; Tribello et al., 2012). These data-driven (as opposed to insight-driven) CVs provide a mechanistic and collective description of the motions. They make use of nonlinear manifold learning techniques to detect hidden nonlinear variables that compactly describe the conformational flexibility of the molecule. Yet, these methods are still limited in scope and cumbersome.

Our goal is to develop computational data-driven perception methods to automatically extract the smooth CVs that best capture the variability of the configurational space, which can be fed directly to standard enhanced sampling methods. The initial data about the conformational variability of the molecule can be based on a variety of phase-space exploration simulation techniques, or on experimentally refined ensembles from nuclear magnetic resonance (Fenwick et al., 2011). Here, we represent in a low-dimensional compact way the reduced slow manifold where the system essentially moves, at the expense of requiring this representation to be nonlinear. We expect that the additional cost will be out-weighted by the additional dimension reduction. We will focus specifically on molecular dynamics simulations, which take a large part of the computer time in supercomputer facilities, often investigating the conformational changes of macromolecules for drug design or disease research. However, the methods we aim at are quite generic, and can be used in other contexts, including real-time simulation and control or optimization of complex engineered systems, see Section 5.

A nonlinear CVs is a smooth function, which assigns to each high-dimensional molecular conformation $r \in \mathbb{R}^D$, a low-dimensional coordinate $\xi \in \mathbb{R}^d$ representative of the state of the system, e.g., during a chemical reaction or a conformational transition:

$$\xi(r) = CV(r).$$

See Fig. 32 for an illustration. For it to be effective, both in explaining the system and in enhancing sampling, it should capture the metastability of the system. In enhanced sampling methods, we need to compute the derivative of the reaction coordinate $\nabla_r CV(r)$. For instance, if we have a potential in the reduced space $U(\xi)$ (e.g. the biasing potential in the metadynamics framework), we can map the collective force to the atomic positions by the chain rule as

$$F_\xi = -\nabla_\xi U, \qquad F_r = F_\xi \, \nabla_r CV.$$

7.2 Examples

We show the current progress of our work in two different molecules, alanine dipeptide and titin. The former is a small dipeptide that serves as benchmark in the area of biochemestry to test the robustness and accuracy of new methods trying to automatically identify the CVs. This is a well known system, fully characterized by two dihedral angles. We test our methodology without using this information, which serves as an a posteriori validation. In the other hand, we consider a titin domain, which is a large protein extensively studied over the last decade, that plays a fundamental role in the elastic properties of human muscles. Titin domains have been studied both experimentally and computationally with pulling tests, in which the protein is unfolded under mechanical force. Here, we identify the collective motions in such steered molecular dynamics simulations.

Alanine dipeptide

This small peptide, see Fig 33A is a standard benchmark in the incipient area of automatic identification of nonlinear CVs in biophysics (Ma and Dinner, 2005; Stamati et al., 2010; Ferguson et al., 2011). The CVs are well understood in terms of two dihedral angles (ψ, ϕ), which have been proven to capture the highly nonlinear motions made by this molecule, see Fig 33A,B. Furthermore, the slow manifold is simple, yet non-trivial, with the topology of a doubly periodic torus, see Fig 33C.

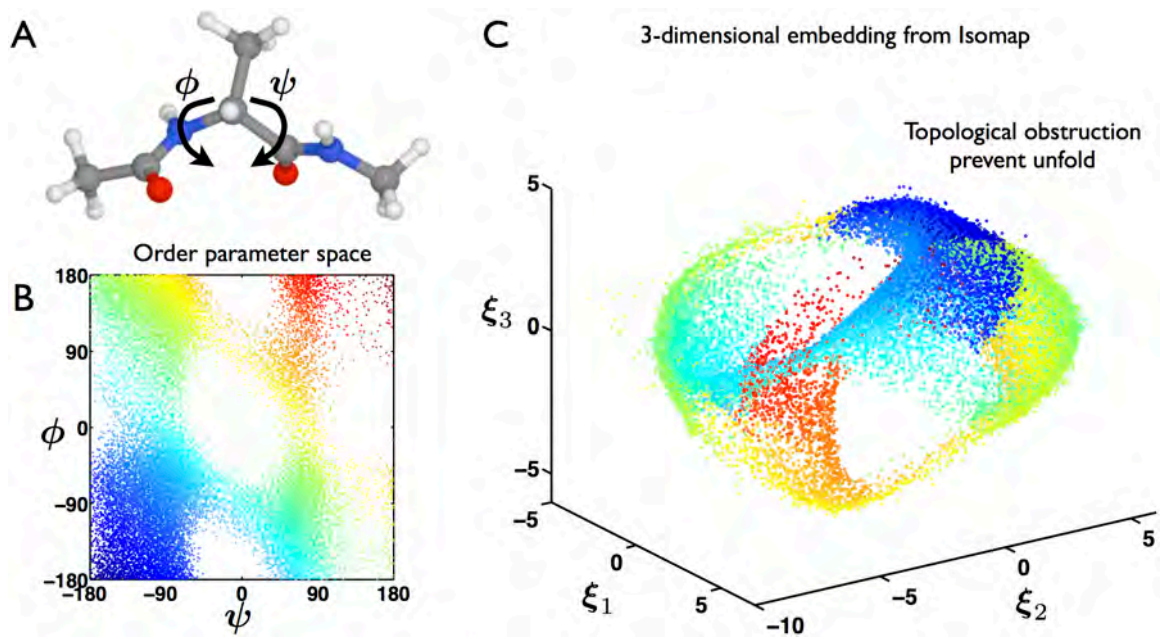


Figure 33: Alanine dipeptide molecule. (A) Molecular representation detailing the main dihedral angles ϕ and ψ. (B) Sampling of the order parameter space from a short time molecular simulation, and (C) 3D embedding of the alanine conformations obtained by Isomap, the alignment has been performed by using the three middle back-bone atoms. Because of the cyclic nature of the coordinates, NLDR techniques cannot unfold the manifold in two dimensions. Manifold tearing is required to overcome this kind of topological obstruction. The color map is a visual guide, and is proportional to the sum of the two dihedral angles.

We explore the configurational space in two steps. (1) A 10 ns long high temperature MD simulation of the NPT ensemble at 573K and 1 bar was performed. A modified Nosé-Hoover method in conjunction with Langevin dynamics were used to control the fluctuations in pressure and temperature. The frequency of sampling was set as 10, leading to 10^6 conformations. (2) A subset of 1400 conformations from the first MD simulation were selected as initial seeds to perform short MD simulations of a NVT ensemble at 310K during 100ps, using Langevin dynamics to control the temperature. The conformations of the first 10ps of these runs were skipped, as the solvent was allowed to equilibrate with a fixed peptide configuration. With a sampling frequency of 20, a total of 4500 conformations for each run were kept at this stage. All the simulations have been performed with NAMD (Phillips et al., 2005), in explicit solvent with TIP3P water molecules, and with a time increment of 1 fs. The alanine dipeptide was modeled with CHARMM22 force field, whereas the electrostatic interaction were evaluated with the Particle Mesh Ewald (PME) method.

The training set to feed the NLDR method was built as follows. First, 1000 conformations from each of the 1400 short MD simulations were chosen, leading to 1,400,000 conformations at 310K. At this stage the representation of each molecule was simplified by removing the hydrogen atoms. Next,

the conformations were aligned by applying procrustes analysis on the 3 middle back-bone atoms, i.e. the middle alpha Carbon atom (gray) and the nearest Nitrogen atom (blue) and Carbon atom, see Fig. 33A. Finally, a decimation of the aligned conformations was carried out such that any two snapshots are at least a cut-off distance away from each other. This procedure gives a final number of 23449 conformations. Their dihedral angles are represented in Fig. 33B. It is clear that, despite the efforts to sample a large portion of configuration space, some regions of high free energy remain unvisited.

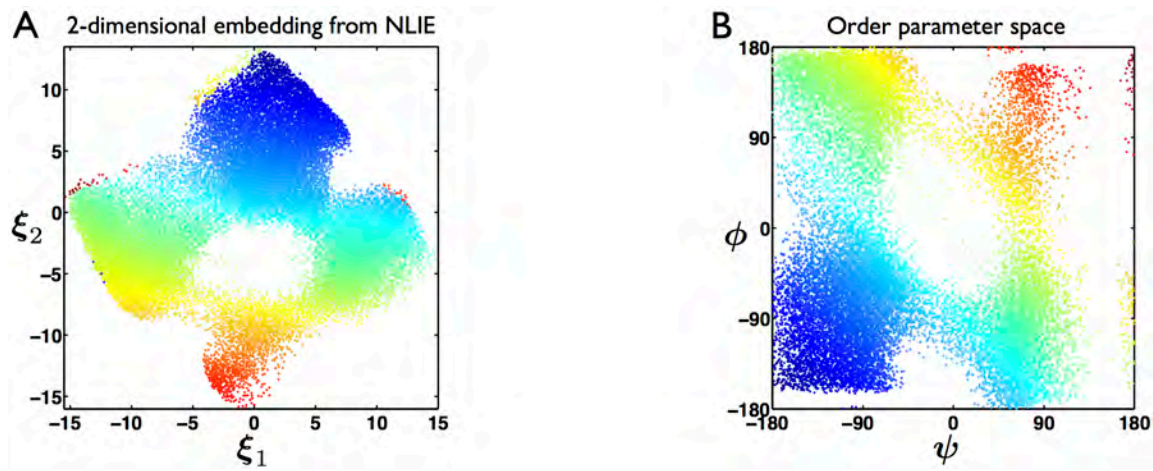


Figure 34: Alanine dipeptide. The topological obstruction that prevents the unfolding can be overcome by removing a few conformations, which can be selected in the 3D embedding given by Isomap, see Fig. 33C. (A) Two-dimensional embedding with NLIE for an ensemble of alanine dipeptide molecules (only back bone atoms). (B) Remaining conformations in dihedral angle space, showing the cut. The color map is a visual guide, proportional to the sum of the two dihedral angles.

To align this molecule, there are two obvious possibilities, either selecting the three middle back-bone atoms or all of them. We first show the results of the manifold processing pipeline for the former case, and then we present the results obtained with the full alignment. Figure 33C depicts the three-dimensional embedding obtained with Isomap. As obvious from the picture, the topology of the slow manifold is non-trivial; it resembles a torus, but slightly more complex due to the interpenetration of one sheet. We tear the high dimensional data by using selected conformations in the three-dimensional embedding, and obtain the expected two-dimensional embedding shown in Fig. 34A. We plot the dihedral angles of the remaining conformations in Fig 34B, where the cuts are evident.

As final step we performed a least squares fitting (LSF) to have a light representation of the slow manifold, see Section 5 for a similar procedure and Appendix B for a concise explanation. The LSF procedure is explained in Figs. 35 and 36, where either the three middle back-bone atoms or all the back-bone atoms are used to align the conformations, respectively. After the fitting, we uniformly sample the two-dimensional embedding and compute the smooth mapping of these positions, see Fig. 35C and Fig. 36C, which finally reveals the hidden slow manifold Fig. 35D and Fig. 36D.

This example shows that, irrespective of the choice of alignment procedure, nonlinear dimensionality reduction methods capture the correct relevant collective variables for a simple, yet nontrivial molecule. The data-based collective variables can be mapped to the two relevant dihedral angles of alanine dipeptide by a simple transformation. They also show that the proposed method is robust, and produces smooth slow manifold representations.

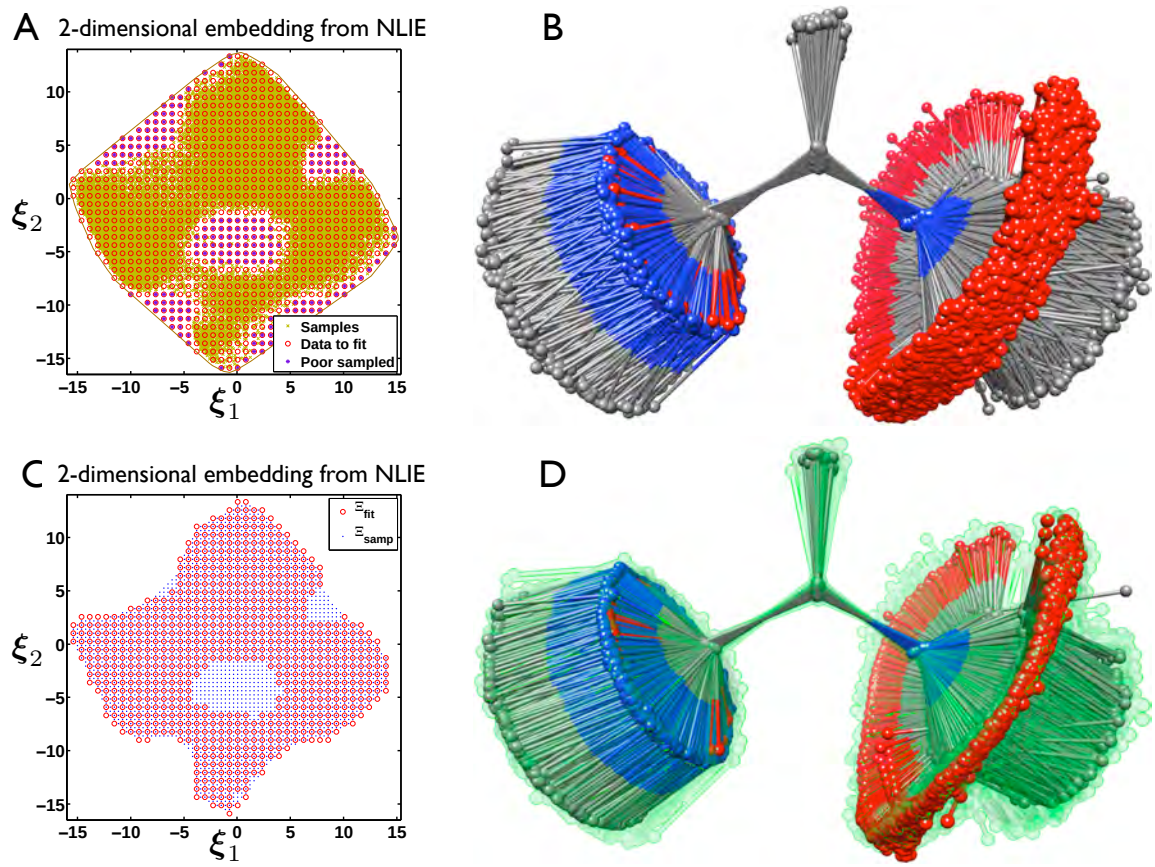


Figure 35: Alanine dipeptide molecule aligned by using the three middle back-bone atoms. (A) Main ingredients involved in the least squares fitting (LSF) of the slow manifold: sample points ($\times$) embedded with the NLIE method, where the LME basis functions are evaluated, control points ($\circ$), which are generated uniformly in the embedded space such that they are contained in a slightly enlarged convex hull of the sample points, and finally the control points with insufficient nearest samples ($*$), that will not be considered in the LSF problem. (B) For concreteness we show few selected high-dimensional conformations of alanine dipeptide (only back bone atoms), used to solve the LSF problem. (C) Embedded space of the alanine dipeptide after fitting: control points ($\circ$), and sample points ($\cdot$) used to explore the smooth manifold. (D) The mapping of the sample points, $\varphi(\xi)$, reveals the slow manifold (only back-bone atoms), and in transparency the noisy conformations used to solve the LSF problem.

Titin I27 domain

Titin is a load-bearing giant muscle protein, extensively studied by force spectroscopy and steered-MD (SMD) over the last 15 years. SMD mimic single molecule experiments. However, the simulation rates are orders of magnitude faster than experimental rates (near equilibrium), which leads to an overestimation of unfolding forces. Still, the unfolding kinematics are accurate (Sotomayor and Schulten, 2007). We consider the titin I27 domain as a proof-of-concept test to study a non well-understood relatively complex molecule. This is actually an important part of the ongoing work in our group.

We explore the configurational space through a SMD simulation. All the simulations have been performed with NAMD (Phillips et al., 2005), in explicit solvent with TIP3P water molecules, and

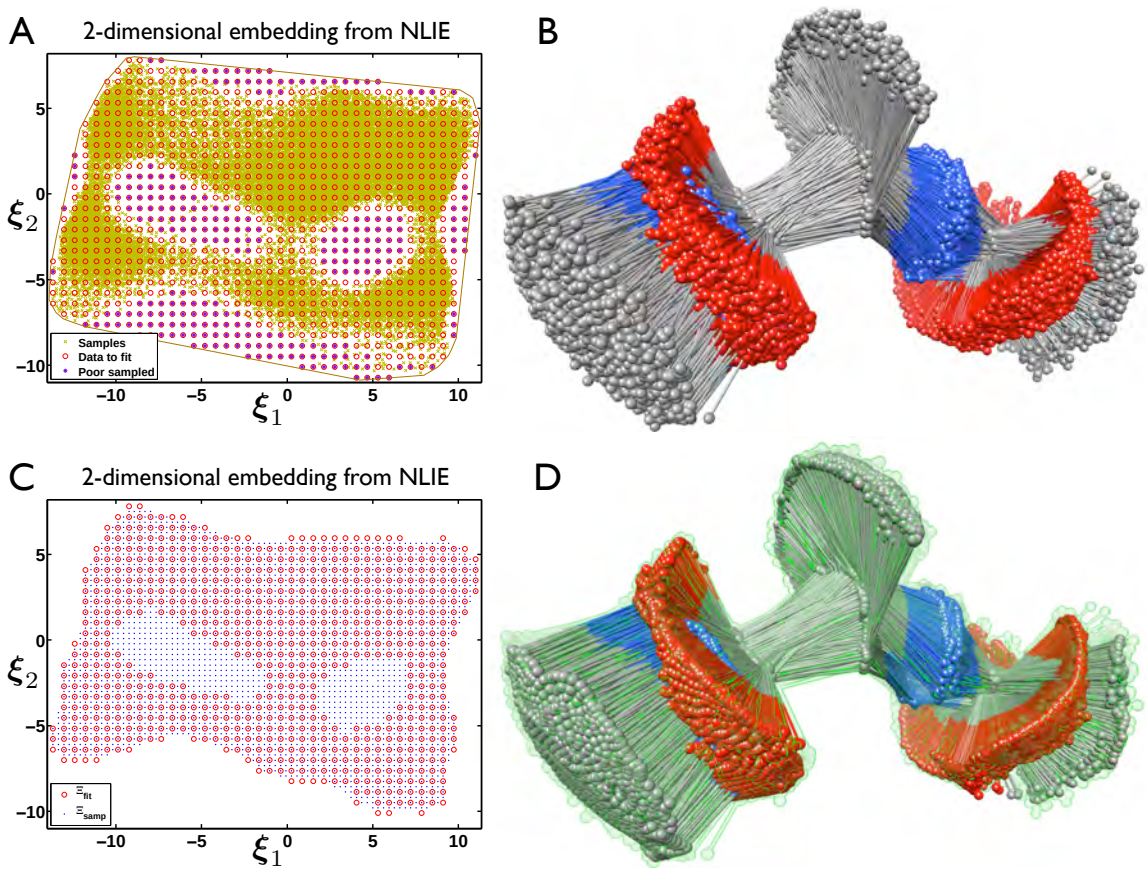


Figure 36: Alanine dipeptide molecule aligned by using all back-bone atoms. (A) Main ingredients involved in the least squares fitting (LSF) of the slow manifold: sample points ($\times$) embedded with the NLIE method, where the LME basis functions are evaluated, control points ($\circ$), which are generated uniformly in the embedded space such that they are contained in a slightly enlarged convex hull of the sample points, and finally the control points with insufficient nearest samples ($*$), that will not be considered in the LSF problem. (B) For concreteness we show few selected high-dimensional conformations of alanine dipeptide (only back bone atoms), used to solve the LSF problem. (C) Embedded space of the alanine dipeptide after fitting: control points ($\circ$), and sample points ($\cdot$) used to explore the smooth manifold. (D) The mapping of the sample points, $\varphi(\xi)$, reveals the slow manifold (only back-bone atoms), and in transparency the noisy conformations used to solve the LSF problem.

with a time increment of 1fs. The system was modeled with CHARMM22 force field, whereas the electrostatic interaction have been evaluated with the PME method. The training set to fed the NLDR methods was built as follows. The representation of each molecule was simplified by considering only the alpha Carbon atoms, removing the hydrogen atoms and other back bones atoms. Next, the conformations were aligned by using procrustes analysis in an incremental procedure, such that the reference conformation has been updated during the alignment. Finally, a decimation of the aligned conformations is carried out such that any two snapshots are at least a cut-off distance away from each other. This procedure gives a final number of 10000 conformations.

Not surprisingly (Hegger et al., 2007), we find that the unfolding dynamics take place along an essentially one-dimensional manifold in conformation space. The CV we find is fundamentally collective, as opposed to a mere end-to-end distance, see Fig. 37.

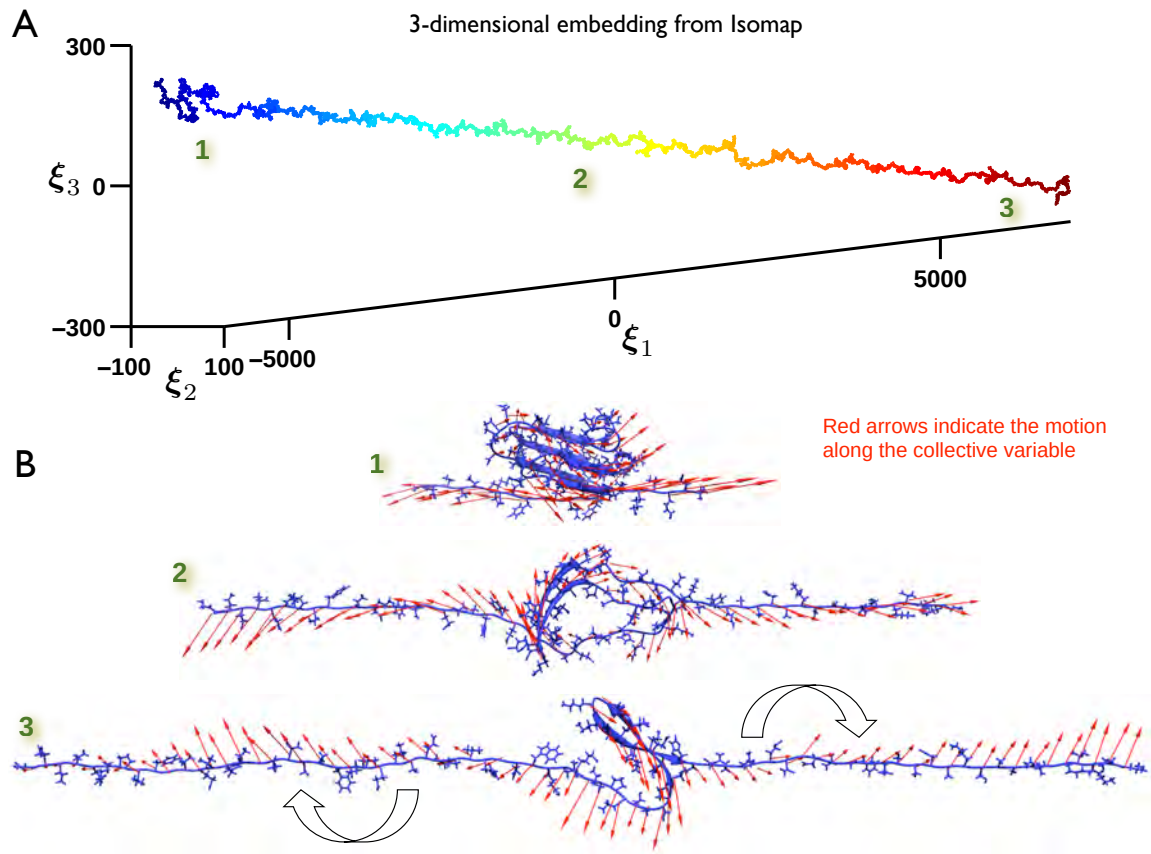


Figure 37: Ongoing research on the collective mechanics of the titin I27 domain. We consider an ensemble of conformations obtained by a SMD simulation. (A) The three-dimensional embedding obtained with Isomap shows, not surprisingly, the one dimensional character of the slow manifold. (B) The gradient of the mapping defined in the one-dimensional embedding is depicted in high dimension for three selected positions, which reveals that the CV is fundamentally collective.

Contributions

We summarize here the most significant contributions of this work, which will be submitted soon (Millán et al., 2012):

1. General method to define smooth CVs from configurational ensembles, based on nonlinear dimensionality reduction.

Open lines for research

We next discuss some open ideas for research derived from the work performed:

(i) We are currently integrating the proposed technique with enhanced sampling methods, which adaptively can erase energy barriers along collective variables. We expect accurate sampling of the thermodynamics with detailed molecular models and moderate computational cost.

(ii) Due to the Gauss Egregium theorem, the low-dimensional embedding of highly curved manifolds cannot be isometric. A parametrization built in a low-dimensional embedding that does

not preserve the isometry changes density of states, which in turn affects the thermodynamic observables. Therefore, a general technique to overcome this problem may consider a partitioning of the high dimensional data, such that the low-dimensional parametrizations are as isometric as possible.

(iii) Similarly to PCV, we can consider the distance to the slow manifold as an extra CV, and assess the metastability in the transversal dimensions. If the selected slow manifold captures the metastability (complexity) of the molecule, then we expect the free energy to be a simple close to quadratic function in the transversal direction. Different features indicate where a more complex description of the molecule is required.

(iv) We expect to continue an ongoing collaboration with the experimental research group led by Dr Salvatella at the Institute for Biomedical Research in Barcelona, in which we are using statistical learning methods to extract hidden information about the conformational dynamics from nuclear magnetic resonance data.

(v) At present, a battery of CVs are offered by almost every well-known molecular dynamics code, including Amber (Cornell et al., 1995), NAMD (Phillips et al., 2005), and GROMACS (Hess et al., 2008), to mention a few. Actually, given a molecular system, it is possible accelerate MD simulations to calculate free energies with these CVs through PLUMED (Bonomi et al., 2009), an open-source plugin under the terms of the GNU Lesser General Public License (LGPL). PLUMED focuses on biological problems, using state-of-the-art methods such as metadynamics, umbrella sampling and Jarzynski's identity-based steered MD. Our purpose is to integrate our methodology in PLUMED, which will allow us to explore the capabilities of the method in a wide range of systems and make it available to the scientific community.

Bibliography

Alexa, M., J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. Silva (2001). Point set surfaces. In *VIS '01: Proceedings of the conference on Visualization '01*, Washington, DC, USA, pp. 21–28. IEEE Computer Society.

Alexa, M., J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. Silva (2003). Computing and rendering point set surfaces. *Transactions on Visualization and Computer Graphics 9*(1), 3–15.

Alexa, M., M. Gross, M. Pauly, H. Pfister, M. Stamminger, and M. Zwicker (2004). Point-based computer graphics. SIGGRAPH 2004 Course Notes.

Amenta, N. and Y. Kil (2004). Defining point-set surfaces. *ACM Transactions on Graphics 23*(3), 264–270.

Amsallem, D., J. Cortial, K. Carlberg, and C. Farhat (2009). A method for interpolating on manifolds structural dynamics reduced-order models. *International Journal for Numerical Methods in Engineering 80*(9), 1241–1258.

Amsallem, D. and C. Farhat (2011). An online method for interpolating linear parametric reduced-order models. *SIAM Journal of Scientific Computing 33*, 2169–2198.

Arroyo, M., L. Heltai, D. Millán, and A. D. and (2012). Reverse engineering the euglenoid swimming stroke. *Proceedings of the National Academy of Sciences of USA*. In press.

Arroyo, M. and M. Ortiz (2006). Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods. *International Journal for Numerical Methods in Engineering 65*(13), 2167–2202.

Bazilevs, Y., V. Calo, J. Cottrell, J. Evans, T. Hughes, S. Lipton, M. Scott, and T. Sederberg (2010). Isogeometric analysis using T-Splines. *Computer Methods in Applied Mechanics and Engineering 199*(5–8), 229–263.

Belkin, M. and P. Niyogi (2003, June). Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation 15*, 1373–1396.

Belytschko, T., H. Stolarski, W. Liu, N. Carpenter, and J. Ong (1985). Stress projection for membrane and shear locking in shell finite-elements. *Computer Methods in Applied Mechanics and Engineering 51*, 221–258.

Bonomi, M., D. Branduardi, G. Bussi, C. Camilloni, D. Provasi, P. Raiteri, D. Donadio, F. Marinelli, F. Pietrucci, R. Broglia, and M. Parrinello (2009). PLUMED: a portable plugin for free-energy calculations with molecular dynamics. *Computer Physics Communications 180*(10), 1961–1972.

Bourdin, B., G. Francfort, and J.-J. Marigo (2008). The Variational Approach to Fracture. *Journal of Elasticity 91*(1-3), 5–148.

Branduardi, D., F. Gervasio, and M. Parrinello (2007, February). From A to B in free energy space. *The Journal of chemical physics 126*(5), 054103.

Brown, W., S. Martin, S. Pollock, E. Coutsias, and J.-P. Watson (2008). Algorithmic dimensionality reduction for molecular structure analysis. *The Journal of Chemical Physics 129*(6), 064118.

Bucalem, M. and J. Bathe (1993). Higher-order mitc general shell elements. *International Journal for Numerical Methods in Engineering 36*(21), 3729–3754.

Ceriotti, M., G. Tribello, and M. Parrinello (2011, July). Simplifying the representation of complex free-energy landscapes using sketch-map. *Proceedings of the National Academy of Sciences of the United States of America*, 1108486108–.

Chang, H. and D.-Y. Yeung (2006, June). Robust locally linear embedding. *Pattern Recognition 39*(6), 1053–1065.

Chen, J. and D. Wang (2006). A constrained reproducing kernel particle formulation for shear deformable shell in cartesian coordinates. *International Journal for Numerical Methods in Engineering 68*(2), 151–172.

Chinesta, F., A. Ammar, and E. Cueto (2010). Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. *Archives for Numerical Methods in Engineering 17*, 327–350.

Choi, H. and S. Choi (2007, March). Robust kernel isomap. *Pattern Recognition 40*(3), 853–862.

Ciarlet, P. (2000). *Mathematical elasticity, Vol III: theory of shells*. North-Holland.

Cirak, F. and Q. Long (2011). Subdivision shells with exact boundary control and non-manifold geometry. *International Journal for Numerical Methods in Engineering 88*(9), 897–923.

Cirak, F. and M. Ortiz (2001). Fully C^1-conforming subdivision elements for finite deformation thin-shell analysis. *International Journal for Numerical Methods in Engineering 51*(7), 813–833.

Cirak, F., M. Ortiz, and P. Schröder (2000). Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *International Journal for Numerical Methods in Engineering 47*(12), 2039–2072.

Conn, A., N. Gould, and P. Toint (1991). A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds. *SIAM J. Numer. Anal. 28*(2), 545–572.

Cornell, W., P. Cieplak, C. Bayly, I. Gould, K. Merz, D. Ferguson, D. Spellmeyer, T. Fox, J. Caldwell, and P. Kollman (1995). A Second Generation Force Field for the Simulation of Proteins, Nucleic Acids, and Organic Molecules. *Journal of the American Chemical Society 117*(19), 5179–5197.

Cottrell, J., T. Hughes, and Y. Bazilevs (2009). *Isogeometric Analysis: Toward Integration of CAD and FEA*. John Wiley & Sons, Ltd.

Cottrell, J., A. Reali, Y. Bazilevs, and T. Hughes (2006). Isogeometric analysis of structural vibrations. *Computer Methods in Applied Mechanics and Engineering 195*(41–43), 5257–5296.

Cox, T. and M. Cox (2001). *Multidimensional scaling* (Second ed.). Boca Raton, FL, USA: Chapman & Hall.

Cyron, C., K. Nissen, V. Gravemeier, and W. Wall (2010). Information flux maximum-entropy approximation schemes for convection-diffusion problems. *International Journal for Numerical Methods in Fluids 64*, 1180–1200.

Darve, E., D. Rodríguez-Gómez, and A. Pohorille (2008, April). Adaptive biasing force method for scalar and vector free energy calculations. *The Journal of chemical physics 128*(14), 144120.

Das, P., M. Moll, H. Stamati, L. Kavraki, and C. Clementi (2006, June). Low-dimensional, free-energy landscapes of protein-folding reactions by nonlinear dimensionality reduction. *Proceedings of the National Academy of Sciences of the United States of America 103*(26), 9885–9890.

de Ridder, D. and R. Duin (2002). Locally linear embedding for classification. Technical Report PH-2002-0, Department of Imaging Science & Technology, Delft University of Technology, Delft, The Netherlands.

de Silva, V. and J. Tenenbaum (2003). Global versus local methods in nonlinear dimensionality reduction. In *Advances in Neural Information Processing Systems 15*, pp. 705–712. MIT Press.

do Carmo, M. (1976). *Differential geometry of curves and surfaces*. Prentice-Hall.

Donoho, D. and C. Grimes (2003, May). Hessian eigenmaps: locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Sciences 100*(10), 5591–5596.

Engel, G., K. Garikipati, T. Hughes, M. Larson, L. Mazzei, and R. Taylor (2002). Continuous/discontinuous finite element approximations of fourth-order elliptic problems in structural and continuum mechanics with applications to thin beams and plates, and strain gradient elasticity. *Computer Methods in Applied Mechanics and Engineering 191*, 3669–3750.

Español, P. (2003). *Statistical Mechanics of Coarse-Graining*. Lecture Notes in Physics. Springer-Verlag.

Fenwick, R., S. Esteban-Martín, B. Richter, D. Lee, K. Walter, D. Milovanovic, S. Becker, N. Lakomek, C. Griesinger, and X. Salvatella (2011). Weak long-range correlated motions in a surface patch of ubiquitin involved in molecular recognition. *Journal of the American Chemical Society 133*(27), 10336–10339.

Ferguson, A., A. Panagiotopoulos, P. Debenedetti, and I. Kevrekidis (2010, August). Systematic determination of order parameters for chain dynamics using diffusion maps. *Proceedings of the National Academy of Sciences of the United States of America 107*(31), 13597–13602.

Ferguson, A., A. Panagiotopoulos, P. Debenedetti, and I. Kevrekidis (2011, April). Integrating diffusion maps with umbrella sampling: application to alanine dipeptide. *The Journal of chemical physics 134*(13), 135103.

Fernández-Méndez, S. and A. Huerta (2004). Imposing essential boundary conditions in mesh-free methods. *Computer Methods in Applied Mechanics and Engineering 193*, 1257–1275.

Fodor, I. (2002). A survey of dimension reduction techniques. Technical Report UCRL-ID-148494, Lawrence Livermore National Laboratory, Livermore, California.

Francfort, G. and J.-J. Marigo (1998). Revisiting brittle fracture as an energy minimization problem. *Journal of the Mechanics and Physics of Solids 46*, 1319–1342.

Gámez, A., C. Zhou, A. Timmermann, and J. Kurths (2004). Nonlinear dimensionality reduction in climate data. *Nonlinear Processes in Geophysics 11*, 393–398.

González, D., E. Cueto, and M. Doblaré (2010). A higher-order method based on local maximum entropy approximation. *International Journal for Numerical Methods in Engineering 83*(6), 741–764.

Green, S. and G. Turkiyyah (2004). Second-order accurate constraint formulation for subdivision finite element simulation of thin shells. *International Journal for Numerical Methods in Engineering 61*(3), 380–405.

Gupta, M. (2003). *An information theory approach to supervised learning*. Ph. D. thesis, Stanford.

Hegger, R., A. Altis, P. Nguyen, and G. Stock (2007). How complex is the dynamics of peptide folding? *Physical Review Letters 98*(2), 028102.

Hérault, J., C. Jausions-Picaud, and A. Guérin-Dugué (1999). Curvilinear component analysis for high-dimensional data representation: I. theoretical aspects and practical use in the presence of noise. In J. Mira and J. Sánchez-Andrés (Eds.), *Engineering Applications of Bio-Inspired Artificial Neural Networks*, Volume 1607 of *Lecture Notes in Computer Science*, pp. 625–634. Springer Berlin / Heidelberg.

Hérault, J., A. Oliva, and A. Guérin-Dugué (1997, April). Scene categorisation by curvilinear component analysis of low frequency spectra. In M. Verleysen (Ed.), *Proceedings of the 5th European Symposium on Artificial Neural Networks*, Bruges, Belgium, pp. 91–96.

Hess, B., C. Kutzner, D. Van Der Spoel, and E. Lindahl (2008). GROMACS 4: Algorithms for Highly Efficient, Load-Balanced, and Scalable Molecular Simulation. *Journal of Chemical Theory and Computation 4*(3), 435–447.

Hoppe, H., T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle (1992). Surface reconstruction from unorganized points. In *SIGGRAPH '92: Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, New York, NY, USA, pp. 71–78. ACM.

Hotelling, H. (1933). Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology 24*(7), 498–520.

Hughes, T., J. Cottrell, and Y. Bazilevs (2005). Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering 194*, 4135–4195.

Jain, A., R. Duin, and J. Mao (Jan. 2000). Statistical pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence 22*(1), 4–37.

Jarzynski, C. (1997). Equilibrium free-energy differences from nonequilibrium measurements: A master-equation approach. *Physical Review E 56*(5), 5018–5035.

Jolliffe, I. (2002). *Principal Component Analysis* (Second ed.). Springer Series in Statistics. New York, NY, USA: Springer.

Karhunen, K. (1946). Zur spektraltheorie stochastischer prozesse. *Annales Academiae Scientiarum Fennicae 34*, 1–7.

Karypis, G. and V. Kumar (1998). *METIS: Unstructured Graph Partitioning and Sparse Matrix Ordering System* (Version 4.0 ed.). Department of Computer Science and Engineering, University of Minnesota.

Kiendl, J., K.-U. Bletzinger, J. Linhard, and R. W uchner (2009). Isogeometric shell analysis with kirchhoff-love elements. *Computer Methods in Applied Mechanics and Engineering 198*(49–52), 3902–3914.

Krysl, P. and T. Belytschko (1996). Analysis of thin shells by the element-free Galerkin method. *International Journal of Solids and Structures 33*(20-22), 3057–3078.

Krysl, P., S. Lall, and J. E. Marsden (2001). Dimensional model reduction in non-linear finite element dynamics of solids and structures. *International Journal for Numerical Methods in Engineering 51*(4), 479–504.

Kumar, S., J. Rosenberg, D. Bouzida, R. Swendsen, and P. Kollman (1992). The weighted histogram analysis method for free-energy calculations on biomolecules. *Journal of Computational Chemistry 13*(8), 1011–1021.

Laio, A. and M. Parrinello (2002). Escaping free-energy minima. *Proceedings of the National Academy of Sciences of the United States of America 99*(20), 12562–12566.

Lancaster, P. and K. Salkauskas (1981). Surfaces generated by moving least squares methods. *Mathematics of Computation 37*(155), 141–158.

Leander, B. (2008). Euglenida. euglenids or euglenoids. Version 11 September 2008. http://tolweb.org/Euglenida/97461/2008.09.11 in The Tree of Life Web Project, http://tolweb.org/.

Leander, B., R. itek, and M. Farmer (2001). Trends in the evolution of the euglenid pellicle. *Evolution 55*, 2215–2235.

Lee, J. and M. Verleysen (2007). *Nonlinear Dimensionality Reduction*. Information science and statistics. New York, NY, USA: Springer.

Levin, D. (1998). The approximation power of moving least-squares. *Mathematics of Computation 67*(224), 1517–1531.

Levin, D. (2003). Mesh-independent surface interpolation. In H. Brunnett and Mueller (Eds.), *Geometric Modeling for Scientific Visualization*, pp. 37–49. Springer-Verlag.

Levoy, M. and T. Whitted (1985). The use of points as displays primitives. Technical Report TR-85-022, Computer Science Department, University of North Carolina at Chappel Hill.

Lew, A., J. Marsden, M. Ortiz, and M. West (2004). Variational time integrators. *International Journal for Numerical Methods in Engineering 60*(1), 153–212.

Li, A. and E. Dowell (2006). Modal reduction of mathematical models of biological molecules. *J. Comp. Phys.*, 211–262.

Liu, Y., W. Wang, B. Lévy, F. Sun, D.-M. Yan, L. Lu, and C. Yang (2009). On Centroidal Voronoi Tessellation—energy smoothness and fast computation. *ACM Trans. Graph. 28*(4), 1–17.

Loève, M. (1955). *Probability Theory*. University series in higher mathematics. New Jersey: Van Nostrand.

Lorenz, E. (1956, December). Empirical orthogonal functions and statistical weather prediction. Statistical Forecasting Project, Scientific Report 1, MIT, Department of Meteorology, Cambridge, MA, USA.

Lumley, J. (1967). The structure of inhomogeneous turbulent flows. In A. M. Yaglom and V. I. Tatarski (Eds.), *Atmospheric turbulence and radio propagation*, pp. 166–178. Moscow: Nauka.

Ma, A. and A. Dinner (2005). Automatic method for identifying reaction coordinates in complex systems. *The Journal of Physical Chemistry B 109*(14), 6769–6779.

MacNeal, R. and R. Harder (1985). A proposed standard set of problems to test finite element accuracy. *Finite Element in Analysis and Design 1*(1), 3–20.

Maisuradze, G., A. Liwo, and H. Scheraga (2009). Principal component analysis for protein folding dynamics. *Journal of Molecular Biology 385*(1), 312–329.

Marsden, J. and M. West (2001). Discrete mechanics and variational integrators. *Acta Numerica 10*, 357–514.

Meyer, T., C. Ferrer-Costa, A. Pérez, M. Rueda, A. Bidon-Chanal, F. J. Luque, C. A. Laughton, and M.Orozco (2006). Essential dynamics: A tool for efficient trajectory compression and management. *J. Chem. Theor. Comput 2*, 251–258.

Millán, D. (2011). *Geometrically nonlinear meshfree thin-shell analysis in the context of Kirchhoff-Love theory.* http://www.youtube.com/RDanielMillan.

Millán, D. and M. Arroyo (2012). Manifold learning for nonlinear model reduction of solid dynamics. *Computer Methods in Applied Mechanics and Engineering*. . Submitted.

Millán, D., B. Hashemian, C. Braga, and M. Arroyo (2012). Describing molecular conformations with smooth collective variables from nonlinear dimensionality reduction. *Journal of Chemical Physics*. In preparation.

Millán, D., A. Rosolen, and M. Arroyo (2011). Thin shell analysis from scattered points with maximum-entropy approximants. *International Journal for Numerical Methods in Engineering 85*(6), 723–751.

Millán, D., A. Rosolen, and M. Arroyo (2012). Nonlinear manifold learning for meshfree finite deformations thin shell analysis. *International Journal for Numerical Methods in Engineering*. DOI: 10.1002/nme.4403.

Niroomandi, S., I. Alfaro, E. Cueto, and F. Chinesta (2010). Model order reduction for hyperelastic materials. *International Journal for Numerical Methods in Engineering 81*(9), 1180–1206.

Nocedal, J. and S. Wright (1999). *Numerical Optimization*. USA: Springer.

Noels, L. (2009). A discontinuous galerkin formulation of non-linear kirchhoff–love shells. *International Journal for Numerical Methods in Engineering 78*(3), 296–323.

Noels, L. and R. Radovitzky (2008). A new discontinuous Galerkin method for Kirchhoff-Love shells. *Computer Methods in Applied Mechanics and Engineering 197*(33-40), 2901–2929.

Noguchi, H., T. Kawashima, and T. Miyamura (2000). Element free analyses of shell and spatial structures. *International Journal for Numerical Methods in Engineering 47*(6), 1215–1240.

Ohtake, Y., A. Belyaev, M. Alexa, G. Turk, and H. Seidel (2003). Multi-level partition of unity implicits. *ACM Transactions on Graphics (Proc. SIGGRAPH 2003) 22*, 463–470.

Patey, G. and J. Valleau (1975, September). A Monte Carlo method for obtaining the interionic potential of mean force in ionic solution. *Journal of Chemical Physics 63*(6), 2334–2339.

Pauly, M. (2003). *Point primitives for interactive modeling and processing of 3D geometry*. Ph. D. thesis, Federal Institute of Technology (ETH) of Zurich.

Pearson, K. (1901). On lines and planes of closest fit to systems of points in space. *Philosophical Magazine 2*(6), 559–572.

Phillips, J., R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R. Skeel, L. Kalé, and K. Schulten (2005). Scalable molecular dynamics with NAMD. *Journal of Computational Chemistry 26*(16), 1781–1802.

Piegl, L. and W. Tiller (1997). *The NURBS Book*. Springer.

Purcell, E. (1977). Life at low Reynolds numbers. *Am J Phys 45*, 3–11.

Rabczuk, T., P. Areias, and T. Belytschko (2007). A meshfree thin shell method for non-linear dynamic fracture. *International Journal for Numerical Methods in Engineering 72*(5), 524–548.

Rosolen, A. and M. Arroyo (2012). Blending isogeometric analysis and maximum entropy meshfree approximants. *Computer Methods in Applied Mechanics and Engineering*. To be submitted.

Rosolen, A., D. Millán, and M. Arroyo (2008). Solmec: an efficient C++ library to solve linear and nonlinear elasticity problems. In *Books of abstracts*, 7th Workshop on Numerical Methods in Applied Science and Engineering, Vall de Núria, Catalonia, Spain, 9–11 January.

Rosolen, A., D. Millán, and M. Arroyo (2010). On the optimum support size in meshfree methods: a variational adaptivity approach with maximum entropy approximants. *International Journal for Numerical Methods in Engineering 82*(7), 868–895.

Rosolen, A., D. Millán, and M. Arroyo (2012). Second order convex *maximum entropy* approximants with applications to high order PDE. *International Journal for Numerical Methods in Engineering*. Accepted.

Roweis, S. and L. Saul (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science 290*(5500), 2323–2326.

Roweis, S. and L. Saul (2003, December). Think globally, fit locally: unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research 4*, 119–155.

Sammon, J. (1969, May). A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers 18*, 401–409.

Sha, F. and L. Saul (2005). Analysis and extension of spectral methods for nonlinear dimensionality reduction. In *Proceedings of the 22nd international conference on Machine learning*, ICML '05, New York, NY, USA, pp. 784–791. ACM.

Simo, J. and D. Fox (1989). On a stress resultant geometrically exact shell model. Part I: Formulation and optimal parametrization. *Computer Methods in Applied Mechanics and Engineering 72*, 267–304.

Simo, J., D. Fox, and M. Rifai (1989). On a stress resultant geometrically exact shell model. Part II: the linear theory; computational aspects. *Computer Methods in Applied Mechanics and Engineering 73*, 53–92.

Sotomayor, M. and K. Schulten (2007). Single-molecule experiments in vitro and in silico. *Science 316*(5828), 1144–1148.

Spiwok, V. and B. Králová (2011). Metadynamics in the conformational space nonlinearly dimensionally reduced by Isomap. *The Journal of Chemical Physics 135*(22), 224504.

Spiwok, V., P. Lipovová, and B. Králová (2007). Metadynamics in essential coordinates: free energy simulation of conformational changes.

Stamati, H., C. Clementi, and L. Kavraki (2010). Application of nonlinear dimensionality reduction to characterize the conformational landscape of small peptides. *Proteins 78*(2), 223–35.

Sukumar, N. (2004). Construction of polygonal interpolants: a maximum entropy approach. *International Journal for Numerical Methods in Engineering 61*(12), 2159–2181.

Suzaki, T. and R. Williamson (1985). Euglenoid movement in euglena fusca: Evidence for sliding between pellicular strips. *Protoplasma 124*, 137–146.

Suzaki, T. and R. Williamson (1986). Cell surface displacement during euglenoid movement and its computer simulation. *Cell Motil Cytoskel 6*, 186–192.

Tama, F. and Y.-H. Sanejouand (2001). Conformational change of proteins arising from normal mode calculations. *Protein Engineering 14*(1), 1–6.

Tenenbaum, J., V. de Silva, and J. Langford (2000). A global geometric framework for nonlinear dimensionality reduction. *Science 290*(5500), 2319–2323.

Torgerson, W. (1952). Multidimensional scaling: I. theory and method. *Psychometrika 17*, 401–419.

Tribello, G., M. Ceriotti, and M. Parrinello (2012, April). Using sketch-map coordinates to analyze and bias molecular dynamics simulations. *Proceedings of the National Academy of Sciences of the United States of America 109*(14), 5196–5201.

Turk, G. and M. Levoy (1994). Zippered polygon meshes from range images. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, SIGGRAPH '94, New York, NY, USA, pp. 311–318. ACM.

van der Maaten, L. J. P., E. O. Postma, and H. J. van den Herik (2009, October). Dimensionality reduction: A comparative review. Technical Report TiCC-TR 2009-005, Tilburg University.

Vanderplas, J. and A. Connolly (2009). Reducing the dimensionality of data: Locally linear embedding of sloan galaxy spectra. *The Astronomical Journal 138*(5), 1365–1379.

Wells, G. and N. Dung (2007). A C^0 discontinuous Galerkin formulation for Kirchhoff plates. *Computer Methods in Applied Mechanics and Engineering 196*(35-36), 3370–3380.

Wells, G. and N. Dung (2008). Geometrically nonlinear formulation for thin shells without rotation degrees of freedom. *Computer Methods in Applied Mechanics and Engineering 197*(35-36), 3370–3380.

Zhang, Z. and J. Wang (2007). Mlle: Modified locally linear embedding using multiple weights. In B. Schölkopf, J. Platt, and T. Hoffman (Eds.), *Advances in Neural Information Processing Systems 19*, pp. 1593–1600. Cambridge, MA: MIT Press.

Zhang, Z. and H. Zha (2005). Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM Journal of Scientific Computing 26*(1), 313–338.

Appendix A

Paper # 1: "Thin shell analysis from scattered points with *maximum-entropy* approximants"

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2011; **85**:723–751
Published online 2 September 2010 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/nme.2992

Thin shell analysis from scattered points with *maximum-entropy* approximants

D. Millán, A. Rosolen and M. Arroyo*, †

Laboratori de Càlcul Numèric (*LaCàN*), *Departament de Matemàtica Aplicada III* (*MA3*), *Universitat Politècnica de Catalunya* (*UPC*), *Campus Nord UPC-C2*, *E-08034 Barcelona*, *Spain*

SUMMARY

We present a method to process embedded smooth manifolds using sets of points alone. This method avoids any global parameterization and hence is applicable to surfaces of any genus. It combines three ingredients: (1) the automatic detection of the local geometric structure of the manifold by statistical learning methods; (2) the local parameterization of the surface using smooth meshfree (here *maximum-entropy*) approximants; and (3) patching together the local representations by means of a partition of unity. Mesh-based methods can deal with surfaces of complex topology, since they rely on the element-level parameterizations, but cannot handle high-dimensional manifolds, whereas previous meshfree methods for thin shells consider a global parametric domain, which seriously limits the kinds of surfaces that can be treated. We present the implementation of the method in the context of Kirchhoff–Love shells, but it is applicable to other calculations on manifolds in any dimension. With the smooth approximants, this fourth-order partial differential equation is treated directly. We show the good performance of the method on the basis of the classical obstacle course. Additional calculations exemplify the flexibility of the proposed approach in treating surfaces of complex topology and geometry.

Received 16 February 2010; Revised 8 June 2010; Accepted 12 June 2010

KEY WORDS: point-set surfaces; meshfree methods; maximum-entropy approximants; thin shells

1. INTRODUCTION

Over the last years, there has been a growing interest in the computer graphics community on point-based surface processing, which presents attractive features as compared with the conventional mesh-based processing [1, 2]. In mesh-based methods, the mesh serves two useful purposes: it describes the geometry of the surface, and the elements provide local parametric spaces where the shape functions and the local parameterizations of the surface can be defined, and where the required calculations on the surface can be performed, e.g. for thin-shell analysis. In these methods, the mesh generation can be difficult, they are not natural for point-based data, and they seem unpractical for embedded manifolds in high-dimensional spaces. On the other hand, in the absence of a mesh, the notion of a surface defined from a set of scattered nodes becomes difficult to grasp. In particular, as noted in [3], a fundamental difficulty in defining basis functions and performing calculations on an embedded surface, as compared with open sub-sets in the Euclidean space, is the absence in general of a single parametric domain. A simple example is the sphere, which does not admit a single singularity-free parameterization. Mesh-based method, consisting of a collection of local parameterizations from the parent element to the physical elements, does not have

*Correspondence to: M. Arroyo, Laboratori de Càlcul Numèric (LaCàN), Departament de Matemàtica Aplicada III (MA3), Universitat Politècnica de Catalunya (UPC), Campus Nord UPC-C2, E-08034 Barcelona, Spain.
†E-mail: marino.arroyo@upc.edu

any difficulty in this respect at the expense of reduced smoothness across the element boundaries. In meshfree methods, such a natural parametric domain is not available, and the description of surfaces with a topology different to that of an open set on $\mathbb{R}^2$, such as a sphere or a torus, becomes challenging. Even for surfaces homeomorphic to open two-dimensional sets, such as that depicted in Figure 18, the geometric complexity can make it very difficult to produce well-behaved global parameterizations.

In the computer graphics literature, Levoy and Whitted [4] pioneered using points as primitives for geometric modeling and rendering of surfaces. Existing methods for describing a surface from a set of scattered points are generally based on implicit representations of the surface [5, 6]. Moving least-squares (MLS) surfaces are a noteworthy example of point-based implicit surface representation, where the surface is defined as a set of fixed points of suitable projections [3, 7]. This idea has been very successful in the computer graphics community for rendering, either up or down-sampling, and manipulating point-set models, see e.g. [1, 2, 8–10]. Despite the common themes and challenges, these developments have remained largely unconnected to the computational mechanics community. In this field, meshfree methods have been applied to thin-shell analysis, and the difficulty of defining an appropriate parametric space has been overcome by considering either a support mesh or very simple surfaces admitting a single parametric space [11–14].

Thin-shell theory requires the approximation of the deformation field to have second-order square integrable derivatives. For general unstructured meshes, it is very difficult to construct C^1 finite element approximations, which has prompted many techniques that avoid this requirement. Most alternative approaches requiring only C^0 elements are based on either Reissner–Mindlin theories for thick plates and shells or on hybrid and mixed methods [15–17]. Excellent reviews and insightful remarks can be found in these references or in [18, 19]. For an extended review, the reader is referred to [20].

Following ideas from computer graphics, Cirak *et al.* [18, 21] presented a direct numerical treatment of thin-shell analysis based on the smooth approximants provided by subdivision surfaces technology. Subdivision finite elements represent unified framework for geometric modeling and thin-shell analysis. On the other hand, discontinuous Galerkin (DG) formulations have been proposed recently for plates, beams and Kirchhoff–Love thin-shells [22–24]. These methods avoid the C^1 continuity requirement by designing suitable numerical fluxes conjugate to the deformation jumps. An advantage of this method is the ease in the imposition of the rotation essential boundary conditions. As disadvantages, the formulation and implementation of these methods is cumbersome, and they typically exhibit a poorer accuracy for a given number of degrees of freedom as compared with methods based on smooth approximants.

Here, we present a method to perform numerical calculations on smooth manifolds described in terms of a set of scattered points. The proposed method avoids any global parametric domain, required in previous meshfree methods, which greatly expands its range of applicability. Although we exercise here the method for surfaces in $\mathbb{R}^3$, it is applicable to perform calculations on embedded manifolds in any space dimension, unlike mesh-based methods. The implications of this attribute of the method are hinted in the closing remarks. The method results from combining three ingredients. First, the local geometric structure of the manifold is detected from the node set using statistical learning methods. Here a weighted Principal Component Analysis (wPCA) identifies the (hyper-) plane closest to the points in a given neighborhood that we call patch. This plane is then used as the local parametric space to construct the meshfree maximum-entropy (*max-ent*) basis functions [25, 26] and the local smooth parameterization of the manifold, and it can be seen as the analogous of the parent element in finite element methods. This smooth parameterization in each patch can be realized with a variety of methods, from other mesh-free methods such as MLS approximants to mesh-based methods such as subdivision finite elements. In the latter case, no global mesh is required. Here the local *max-ent* approximants [25] are chosen, due to their smoothness, robustness, and the relative ease of quadrature relative to other meshfree approximants. The different local parameterizations are then glued together with a partition of unity (PU) defined in the ambient space, which consequently is also a PU in the embedded manifold. Specifically, functionals defined on the surface are readily split into local contributions, each involving a single local parameterization.

DOI: 10.1002/nme

The outline of the paper is as follows. Section 2 describes the proposed methodology for point-set manifold processing. This methodology is illustrated with the simple yet illustrative calculation of the Willmore energy of a sphere. Next, Section 3, provides a short account of the Kirchhoff–Love shell theory. Throughout the present work, we confine our attention to the linear theory of shells under static loading. Numerical experiments to evaluate the performance of the method are presented in Section 4, with particular attention to the obstacle course of benchmark tests proposed by Belytschko *et al.* [27]. Other numerical examples illustrate the ability of the method to treat shells with complex topology or geometry. Some remarks and conclusions are collected in Section 5.

2. MANIFOLD DESCRIPTION FROM SCATTERED POINTS

We consider a smooth d-manifold $\mathscr{M}$ embedded in $\mathbb{R}^D$, $d \leqslant D$. Our aim is to obtain a numerical representation of $\mathscr{M}$, and make computations on it. Manifolds are important objects in mathematics and physics because they allow us to understand complicated structures in each neighborhood in terms of Euclidean spaces. The first of the three steps in the methodology presented here precisely captures numerically the local Euclidean structure of the manifold in a set of neighborhoods that we call patches.

Let $P = \{\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_N\} \subset \mathbb{R}^D$ be a set of points representing $\mathscr{M}$. We detect the local Euclidean structure and build local parameterizations around the patch points, $Q = \{\boldsymbol{Q}_1, \boldsymbol{Q}_2, \ldots, \boldsymbol{Q}_M\}$, typically a subset of P but not necessarily. For simplicity, we will denote the points in P and its associated objects with a lower case subindex, e.g. $\boldsymbol{P}_a$, for $a = 1, 2, \ldots, N$, and the patch points in Q and its associated objects with an upper case subindex, e.g. $\boldsymbol{Q}_A$, for $A = 1, 2, \ldots, M$.

We recall the construction of a Shepard PU, for later reference. Let $Y = \{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_L\} \subset \mathbb{R}^D$ be a set of points and consider a set of non-negative reals $\{\beta_a\}_{a=1,2,\ldots,L}$ associated with each point in Y. We define the Shepard PU with the Gaussian weight related to the set Y as the functions $w_a^Y : \mathbb{R}^D \to \mathbb{R}$ for $a = 1, 2, \ldots, L$ given by

$$w_a^Y(\boldsymbol{x}) = \frac{\exp(-\beta_a |\boldsymbol{x} - \boldsymbol{y}_a|^2)}{\sum_{b=1}^{L} \exp(-\beta_b |\boldsymbol{x} - \boldsymbol{y}_b|^2)}. \tag{1}$$

For efficiency, and given the fast decay of the Gaussian functions, it is convenient to consider numerically these functions to be compactly supported. We define for each point $\boldsymbol{x}$ the neighborhood index set from Y as

$$\mathscr{N}_{\boldsymbol{x}}^Y = \{a \in \{1, 2, \ldots, L\} \,|\, \exp(-\beta_a |\boldsymbol{x} - \boldsymbol{y}_a|^2) > TOL\}, \tag{2}$$

where TOL is a numerical tolerance. Then, the evaluation of the Shepard functions becomes

$$w_a^Y(\boldsymbol{x}) = \begin{cases} \dfrac{\exp(-\beta_a |\boldsymbol{x} - \boldsymbol{y}_a|^2)}{\sum_{b \in \mathscr{N}_{\boldsymbol{x}}^Y} \exp(-\beta_b |\boldsymbol{x} - \boldsymbol{y}_b|^2)} & \text{for } a \in \mathscr{N}_{\boldsymbol{x}}^Y, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

A dual notion of neighborhood index list is given by the points in a given set $Z = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_O\}$ that lie within the support of a given Shepard function w_a^Y, i.e.

$$\hat{\mathscr{N}}_{\boldsymbol{y}_a}^Z = \{b \in \{1, 2, \ldots, O\} \,|\, w_a^Y(\boldsymbol{z}_b) > TOL\}. \tag{4}$$

Following [25], we define a dimensionless parameter $\gamma_a = \beta_a h_a^2$, the *aspect ratio parameter* of the weighting or shape functions, where h_a is a measure of the point spacing around $\boldsymbol{y}_a$. As the value of γ increases, the Shepard functions become narrower. For a systematic method for defining the typical point spacing, see [10].

DOI: 10.1002/nme

In the following, we will consider Shepard functions associated with the point-set P, which will be denoted as $w_a^P(\boldsymbol{x}), a=1,2,\dots N$, and with the set of patch points Q, denoted by $w_A^Q(\boldsymbol{x})$, $A=1,2,\dots M$. Note that each of these two PUs are defined not only from different node sets, but also from different sets of aspect ratio parameters. We also note that each of these sets of functions sums to one in every point in $\mathbb{R}^D$.

2.1. Local Euclidean structure

We turn now to a fundamental element in the proposed method, that of computing the local geometric structure of the manifold around $\boldsymbol{Q}_A$. For this purpose, we use a fundamental idea in statistical learning: principal component analysis (PCA). PCA is a standard tool in computer graphics [2], data analysis [28], manifold learning [29], or model reduction techniques in computational mechanics based on proper orthogonal decomposition (POD) methods [30, 31]. PCA identifies the low-dimensional subspace that best explains the variance of a higher-dimensional data set. The original data are transformed into a new orthogonal coordinate system such that the projection of the data on the subspace defined by the first m coordinate directions maximizes the variance as compared with any other projection onto an m-dimensional subspace.

To describe the manifold properties around a patch point $\boldsymbol{Q}_A$, a weighted local version, which we call wPCA (also kernel PCA), is more suitable. The main idea is to define a small neighborhood around $\boldsymbol{Q}_A$, and compute the covariance matrix of the points of P in this neighborhood. If the local structure of the point-set around $\boldsymbol{Q}_A$ is that of a d-manifold, then the first d eigenvalues of the covariance matrix will strongly dominate the spectrum. If this is not the case, it is possible that the noise in the data is too large, and a larger domain of influence of the weighting function may be adequate. If this domain of influence is too large, the drawback is a loss of the local features of the manifold.

The average of the points in the neighborhood of $\boldsymbol{Q}_A$ is then

$$\bar{\boldsymbol{Q}}_A = \sum_{a\in\mathcal{N}^P_{\boldsymbol{Q}_A}} w_a^P(\boldsymbol{Q}_A)\boldsymbol{P}_a,$$

where $\mathcal{N}^P_{\boldsymbol{Q}_A}$ denotes the neighbor index set of $\boldsymbol{Q}_A$ relative to the Shepard functions of P. Arranging the points as column vectors, we define the matrix $\boldsymbol{X}_A \in \mathbb{R}^{D\times \mathrm{card}(\mathcal{N}^P_{\boldsymbol{Q}_A})}$ whose columns are $\boldsymbol{P}_a - \bar{\boldsymbol{Q}}_A$, for $a\in\mathcal{N}^P_{\boldsymbol{Q}_A}$. The covariance matrix is then

$$\boldsymbol{C}_A = \boldsymbol{X}_A \,\mathrm{diag}\{w_a^P(\bar{\boldsymbol{Q}}_A) | a\in\mathcal{N}^P_{\boldsymbol{Q}_A}\}(\boldsymbol{X}_A)^{\mathrm{T}} \in \mathbb{R}^{D\times D}.$$

This positive- (semi-) definite symmetric matrix has real eigenvalues and diagonalizes in an orthonormal basis of eigenvectors. We define $\boldsymbol{V}_A \in \mathbb{R}^{D\times d}$ as the eigenvector matrix formed by the d eigenvectors corresponding to the largest d eigenvalues. These eigenvectors form an orthogonal basis of the local Euclidean structure of the manifold, which can be informally thought of as a numerical tangent space at $\boldsymbol{Q}_A$. This plane, T_A (see Figure 1), passes through $\bar{\boldsymbol{Q}}_A$ and is parallel to these first d eigenvectors. The matrix $\boldsymbol{V}_A$ defines an orthogonal projection relative to $\bar{\boldsymbol{Q}}_A$ onto the reduced space of dimension d, i.e.

$$\begin{aligned}\Pi_A : \mathbb{R}^D &\longrightarrow \mathbb{R}^d \\ \boldsymbol{z} &\longmapsto (\boldsymbol{V}_A)^{\mathrm{T}}(\boldsymbol{z}-\bar{\boldsymbol{Q}}_A).\end{aligned}$$

Note that in general $\bar{\boldsymbol{Q}}_A \neq \boldsymbol{Q}_A$; hence $\Pi_A(\boldsymbol{Q}_A)\neq \boldsymbol{0}$.

2.2. Local parameterization

The local Euclidean structure around $\boldsymbol{Q}_A$ provides a convenient parametric space for the embedded manifold, as is shown in Figure 2. Indeed, consider the reduced dimensionality point-set

$$\Xi_A = \{\boldsymbol{\xi}_a = \Pi_A(\boldsymbol{P}_a) | a\in\hat{\mathcal{N}}^P_{\boldsymbol{Q}_A}\} \subset \mathbb{R}^d. \tag{5}$$

DOI: 10.1002/nme

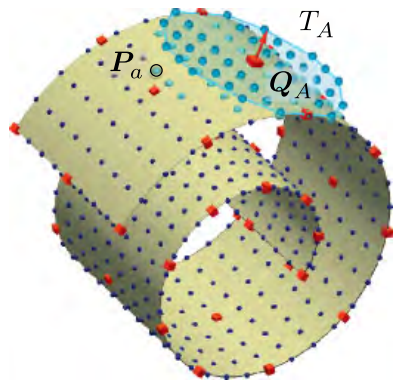


Figure 1. Local Euclidean structure (numerical tangent plane) T_A at the patch point $\boldsymbol{Q}_A$ (box) obtained by applying wPCA. The small spheres represent the N points in P, whereas the big spheres are the points in the neighborhood $\mathcal{N}^P_{\boldsymbol{Q}_A}$ and their projection onto T_A. For this example, the eigenvalues of the covariance matrix are $\lambda = \{0.514, 0.485, 0.001\}$ (arbitrary units). The last eigenvalue is associated with the eigenvector normal to T_A.

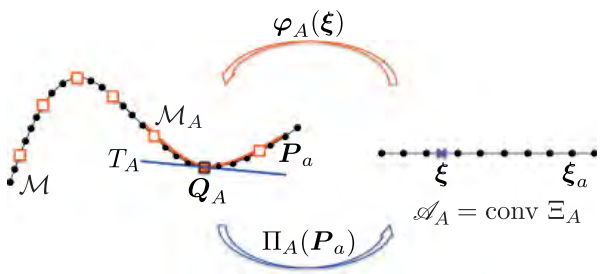


Figure 2. Illustration of the local parameterization $\boldsymbol{\varphi}_A$ of $\mathscr{M}$ around $\boldsymbol{Q}_A$. $\mathscr{M}_A$ is a local approximation to the surface in the vicinity of $\boldsymbol{Q}_A$, T_A is the numerical tangent, and $\mathscr{A}_A = \text{conv}\Xi_A$ is the local parametric space. Small solid points represent the N points in P, whereas the open square symbols represent the M patches in Q associated with the local parameterizations. Each local parameterization has an associated function in the PU w^Q_A.

The convenience of choosing this neighborhood list (the points in P lying within the support of w^Q_A) will become clear below. Suppose we have smooth approximants $p_a(\boldsymbol{\xi})$ associated with this point-set on a subset $\mathscr{A}_A$ of $\mathbb{R}^d$, here the convex hull of the projected node set $\text{conv}\Xi_A$, and we locally parameterize the manifold around $\boldsymbol{Q}_A$ as

$$\begin{aligned} \boldsymbol{\varphi}_A : \mathbb{R}^d \supset \text{conv}\Xi_A &\longrightarrow \mathbb{R}^D \\ \boldsymbol{\xi} &\longmapsto \sum_{a \in \hat{\mathcal{N}}^P_{\boldsymbol{Q}_A}} p_a(\boldsymbol{\xi}) \boldsymbol{P}_a . \end{aligned}$$

The image of $\mathscr{A}_A = \text{conv}\Xi_A$ through the parameterization $\boldsymbol{\varphi}_A$, $\mathscr{M}_A$, is a local approximation of the manifold around $\boldsymbol{Q}_A$, see Figure 2.

2.3. PU to evaluate integrals on $\mathscr{M}$

A PU is a classical technique to patch together local constructions on a manifold [32]. We consider now a Shepard PU referred to a set of patch points, i.e. the functions $w^Q_A(\boldsymbol{x})$, $A = 1, 2, \ldots, M$. These functions form a PU in $\mathbb{R}^D$, and consequently also in $\mathscr{M}$. Other partitions of unity are perfectly valid, including the local *max-ent* approximants described below used for the local parameterizations.

DOI: 10.1002/nme

Our choice, the Shepard basis functions, is straightforward to evaluate and allow us to easily control the extent of the support.

Consider now the integral of a scalar function f over a manifold $\mathscr{M}$, $f:\mathscr{M}\to\mathbb{R}$. This function can also depend on other fields, as in a functional over the manifold or its associated weak form. Then we have the following identity:

$$\int_{\mathscr{M}} f(\boldsymbol{x})\,\mathrm{d}_{\mathscr{M}} = \sum_{A=1}^{M}\int_{\mathscr{M}} w_A^Q(\boldsymbol{x})\,f(\boldsymbol{x})\,\mathrm{d}_{\mathscr{M}}. \tag{6}$$

Combining the PU with the local parameterization around $\boldsymbol{Q}_A$, we can approximate numerically integrals over the manifold $\mathscr{M}$ described by a set of scattered points as

$$\int_{\mathscr{M}} f(\boldsymbol{x})\,\mathrm{d}_{\mathscr{M}} \simeq \sum_{A=1}^{M}\int_{\mathscr{A}_A} w_A^Q(\boldsymbol{\varphi}_A(\boldsymbol{\xi}))\,f(\boldsymbol{\varphi}_A(\boldsymbol{\xi}))\,J_A(\boldsymbol{\xi})\,\mathrm{d}\boldsymbol{\xi},$$

where $\mathscr{A}_A$ is the parametric space associated with the Ath patch, and $J_A=\sqrt{\det[(D\boldsymbol{\varphi}_A)^{\mathrm{T}}D\boldsymbol{\varphi}_A]}$ is the Jacobian determinant of the parameterization. In this way, similar to finite element methods, we have split the integral into local contributions that can be evaluated using local parameterizations.

The last integral can be subsequently approximated by numerical quadrature on the local parametric space. Here, we use the Gauss quadrature on a support triangulation defined over Ξ_A. The integrand often does not depend explicitly on $\boldsymbol{x}$, and we express f in terms of $\boldsymbol{\xi}$ directly. This is not the case for the PU functions, which are defined on the ambient space $\mathbb{R}^D$. Defining a PU intrinsically on $\mathscr{M}$ would be useful but seems difficult to implement. Also, from the above expression it is clear that the PU functions need to be compactly supported in $\mathscr{M}_A$, which motivates the choice of neighborhood index set in Equation (5). As a matter of fact, in computations we ensure this by choosing a slightly larger index set in Equation (5), see Figure 3.

We also note that in the present work, we establish a one-to-one correspondence between local parameterizations or patches and PU Shepard functions. This is not a strict requirement, and other possibilities may prove convenient.

2.4. *Maximum-entropy approximants*

The smooth approximants $p_a(\boldsymbol{\xi})$ used here are local *max-ent* (LME) meshfree approximants [25]. We briefly outline their construction and basic properties here. These approximation schemes fall into the class of convex approximants, such as natural neighbor approximants [33], subdivision approximants [18], or NURBS and B-splines basis functions [34]. Convex basis functions, which we

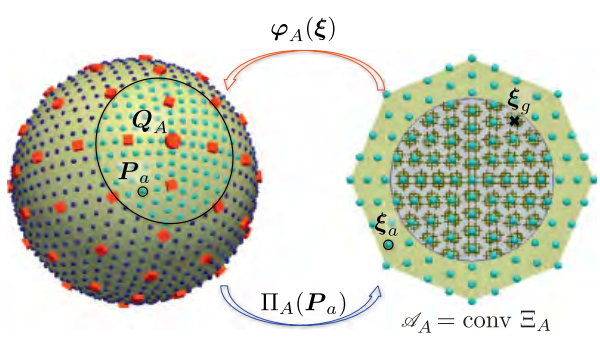


Figure 3. Illustration of the local parameterization $\boldsymbol{\varphi}_A$ of a sphere around a patch point $\boldsymbol{Q}_A$. Small solid spheres represent the N points in P, whereas the box symbols represent the M patches where the local parameterizations are constructed (left). The points $\boldsymbol{\xi}_a=\Pi_A(\boldsymbol{P}_a)$ (the projections of the points $\boldsymbol{P}_a, a\in\hat{\mathscr{N}}_{\boldsymbol{Q}_A}^{P}$ onto the local parametric space $\mathscr{A}_A$) are represented by spheres, whereas brown cross symbols stand for the Gauss points $\boldsymbol{\xi}_g$ within in the support of $w_A^Q\circ\boldsymbol{\varphi}_A$, displayed in gray (right).

DOI: 10.1002/nme

will denote by $p_a(\boldsymbol{\xi}), a=1,\ldots,n$ with $\boldsymbol{\xi}\in\mathbb{R}^d$, are non-negative linearly reproducing approximants intimately related to convex geometry. To simplify the notation in this section, we renumber the point indices for a given patch $\hat{\mathcal{N}}^P_{\mathbf{Q}_A}$, from 1 to $n=\text{card}(\hat{\mathcal{N}}^P_{\mathbf{Q}_A})$.

The local *max-ent* approximants used here are first-order consistent. Extensions to second-order approximation schemes based on adding extra constraints to the convex program below can be found in [26] Rosolen *et al.* (in preparation), whereas González *et al.* [35] propose a method based on the de Boor's algorithm. For a set of nodal values $\{u_a\}_{a=1,\ldots,n}$ associated with a nodal set $\{\boldsymbol{\xi}_a\}_{a=1,\ldots,n}$, the approximation of a function $u(\boldsymbol{\xi})$ is given by

$$u(\boldsymbol{\xi})\approx\sum_{a=1}^{n}p_a(\boldsymbol{\xi})u_a,$$

where the approximants p_a are non-negative and fulfill the zeroth and first-order consistency conditions:

$$p_a(\boldsymbol{\xi})\geqslant 0,\quad \sum_{a=1}^{n}p_a(\boldsymbol{\xi})=1,\quad \sum_{a=1}^{n}p_a(\boldsymbol{\xi})\boldsymbol{\xi}_a=\boldsymbol{\xi}.$$

The definition of the *max-ent* approximants is not explicit, but rather follows from an optimization problem set up at each evaluation point $\boldsymbol{\xi}$, where the unknowns are the values of the basis functions at this point $p_a(\boldsymbol{\xi}), a=1,\ldots,n$, and where the above conditions are seen as constraints on these unknowns. As shown in [25], the constraints are only feasible within the convex hull of the node set.

As the approximants are non-negative and form a PU (zeroth-order consistency condition) at each point $\boldsymbol{\xi}$, they can be interpreted as a probability distribution. This information–theoretical viewpoint allows us to pose a statistical inference problem where the approximants p_a are the unknowns. A canonical measure of the uncertainty associated with a discrete probability distribution is the entropy, and the principle of maximum entropy provides the least biased approximation schemes consistent with the constraints. As proposed in [25], it proves convenient to select a compromise between maximizing the entropy and minimizing the width or support-size of the basis functions, which leads to the following convex program [36]:

$$\begin{aligned}\text{For fixed } \boldsymbol{\xi},\ \text{minimize}\quad & \sum_{a=1}^{n}\beta_a p_a|\boldsymbol{\xi}-\boldsymbol{\xi}_a|^2+\sum_{a=1}^{n}p_a\ln p_a\\ \text{subject to}\quad & p_a\geqslant 0,\quad a=1,\ldots,n\\ & \sum_{a=1}^{n}p_a=1,\quad \sum_{a=1}^{n}p_a\boldsymbol{\xi}_a=\boldsymbol{\xi}.\end{aligned}$$

This convex optimization problem is solved efficiently and robustly with duality methods [25], obtaining

$$p_a(\boldsymbol{\xi})=\frac{1}{Z(\boldsymbol{\xi},\lambda^*(\boldsymbol{\xi}))}\exp[-\beta_a|\boldsymbol{\xi}-\boldsymbol{\xi}_a|^2+\lambda^*(\boldsymbol{\xi})\cdot(\boldsymbol{\xi}-\boldsymbol{\xi}_a)],$$

where

$$Z(\boldsymbol{\xi},\lambda)=\sum_{b=1}^{n}\exp[-\beta_b|\boldsymbol{\xi}-\boldsymbol{\xi}_b|^2+\lambda\cdot(\boldsymbol{\xi}-\boldsymbol{\xi}_b)]$$

is the partition function and λ^* is the solution at each point $\boldsymbol{\xi}$ of the convex and unconstrained optimization problem

$$\lambda^*(\boldsymbol{\xi})=\arg\min_{\lambda\in\mathbb{R}^d}\ln Z(\boldsymbol{\xi},\lambda).\tag{7}$$

These approximants are meshfree since they are entirely defined by the node set $\{\boldsymbol{\xi}_a\}_{a=1,\ldots,n}$ and the parameters $\{\beta_a\}_{a=1,\ldots,n}$. The parameter $\beta_a>0$ is a measure of the locality of the shape

DOI: 10.1002/nme

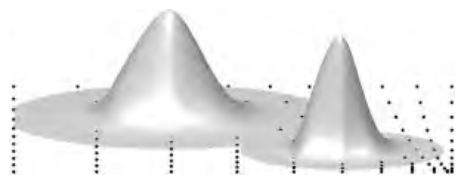


Figure 4. Local *max-ent* approximants for a non-uniform nodal distribution. The locality parameter $\beta_a = \gamma / h_a^2$ at each node is selected to achieve a uniform aspect parameter $\gamma = 1.6$.

functions. As mentioned in Section 2.1, a dimensionless aspect parameter $\gamma_a = \beta_a h_a^2$ characterizes the aspect ratio of the shape functions relative to the typical nodal spacing h_a in the vicinity of $\boldsymbol{\xi}_a$. As the value of γ_a increases, the corresponding shape function is sharper and more local, and in the limit $\gamma_a \to \infty$ coincides with the Delaunay hat shape function. For highly unstructured point-sets, it is easy to produce local *max-ent* approximants of uniform aspect ratio by appropriately selecting a non-uniform value of the parameters β_a, resulting in a uniform value for the parameters γ_a (see Figure 4).

It has been shown that these approximants are C^∞. Their gradients can be computed analytically. The expressions for the first and second spatial derivatives of local *max-ent* approximants are given in Appendix A, see also [36]. These approximants are non-interpolating, except at the boundary of the convex hull of the node set, where a weak Kronecker-delta property holds. This property makes it straightforward to impose essential boundary conditions unlike other meshfree methods such as those based on the MLS approximations [37]. Finally, we note that local *max-ent* approximants deal robustly with training data infected with zero-mean additive noise, as proven for non-parametric supervised classification [38].

2.5. Post-processing of the surface geometry and fields on the surface

The proposed method does not provide a way to patch together the local overlapping parameterizations of the surface or of fields on the surface. Indeed, the patching is performed at the level of the integrals. We describe next the procedure to post-process shapes and displacement fields used here. We assume that a number of points in space are sampling the true surface, that in general does not coincide with the sets P and Q. Given such a point $\boldsymbol{y} \in \mathscr{M}$, the numerical approximation of the surface takes the form

$$\boldsymbol{\varphi}^h(\boldsymbol{y}) = \sum_{A \in \mathscr{N}_{\boldsymbol{y}}^Q} w_A^Q(\boldsymbol{y}) \sum_{a \in \hat{\mathscr{N}}_{Q_A}^P} p_a(\Pi_A(\boldsymbol{y})) \boldsymbol{P}_a, \tag{8}$$

where the patch projection Π_A has been defined in Section 2.1. Similarly, other fields on the surface described in terms of nodal values can be evaluated, for instance the displacement field

$$\boldsymbol{u}^h(\boldsymbol{y}) = \sum_{A \in \mathscr{N}_{\boldsymbol{y}}^Q} w_A^Q(\boldsymbol{y}) \sum_{a \in \hat{\mathscr{N}}_{Q_A}^P} p_a(\Pi_A(\boldsymbol{y})) \boldsymbol{u}_a.$$

We note that in Equation (8), the points $\boldsymbol{P}_a$ can be viewed as control points, following the B-Spline terminology. Indeed, as in B-Splines or NURBS, a point of the numerical surface is obtained as a convex combination of the control points, whose coefficients are the basis functions evaluated at a particular point in parameter space. Also, in the present context the numerical approximation of the surface does not pass through the control points either, and a control point has a local effect on the surface in its vicinity. Thus, in general, the control points should be chosen such that the actual surface, given analytically or through a set of sample points, is accurately reproduced, for instance with a least-square fit. In the examples below, we do not perform this pre-processing step, and directly place the points in P on the surface. This results in a small error in the description of the geometry that disappears as the discretization is refined.

DOI: 10.1002/nme

2.6. Example: *the Willmore energy of a sphere*

The computation of the Willmore energy (related to the curvature strain energy of thin elastic sheets) of a spherical surface is presented next to illustrate and assess the methodology. The expression of the energy is

$$E = \int_{\Omega} H^2 \, \mathrm{d}\Omega, \tag{9}$$

where H is the mean curvature and $d\Omega$ is the area element of the surface Ω. This functional involves the computation and integration of second-order spatial derivatives. For a sphere, we have $E = 16\pi$.

Following the method described above and for a surface described by a set of points $P = \{\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_N\} \subset \mathbb{R}^3$, the elastic energy E is approximated by

$$\begin{aligned} E &= \int_{\Omega} H^2 \, \mathrm{d}\Omega \\ &\simeq \sum_{A=1}^{M} \int_{\mathscr{A}_A} w_A^Q(\boldsymbol{\varphi}_A(\boldsymbol{\xi})) H^2(\boldsymbol{\xi}) J_A(\boldsymbol{\xi}) \, \mathrm{d}\boldsymbol{\xi}, \end{aligned} \tag{10}$$

where for a surface we can compute the Jacobian determinant as $J_A = \|\boldsymbol{\varphi}_{A,1} \times \boldsymbol{\varphi}_{A,2}\|$. In the equation above, H is not an explicit function of $\boldsymbol{\xi}$, but rather of the first and second derivatives of the local parameterization $\boldsymbol{\varphi}_A$. The energy in Equation (10) is thus split into patch contributions $E \simeq E^h = \sum_{A=1}^{M} E_A^h$, which are evaluated using numerical quadrature

$$E_A^h = \sum_{g=1}^{n_g} w_A^Q(\boldsymbol{\varphi}_A(\boldsymbol{\xi}_g)) H^2(\boldsymbol{\xi}_g) J_A(\boldsymbol{\xi}_g) w_g,$$

where $\{\boldsymbol{\xi}_g\}_{g=1,\ldots,n_g}$ are the quadrature point coordinates in the parametric space $\mathscr{A}_A$, with the corresponding weights w_g. We define the relative error for the elastic energy of the sphere as

$$e_{\mathrm{rel}} = \frac{|16\pi - E^h|}{16\pi}.$$

We investigate the influence in the computation of E^h of (i) the accuracy of the integration rule and (ii) some key numerical parameters, see Table I. We consider the Gauss–Legendre cubature rules [39] of order $1, 2, 3, 4, 5, 6, 8, 10$, and 15 for the Delaunay triangulation supported on the projected node sets Ξ_A; these schemes have $1, 3, 4, 6, 7, 12, 16, 25$, and 54 quadrature points per triangle, respectively. The numerical parameters entering the calculation, collected in the table, are a set of tolerances and aspect ratio parameters. More specifically, the tolerance TOL that sets the cutoff of the neighbor search, see Equations (2,4), can be chosen independently for the Shepard functions involved in the wPCA step, w_a^P, for the Shepard functions defining the PU that splits the integrals, w_A^Q, and for the local *max-ent* (LME) approximants p_a. Similarly, the aspect ratio parameter can be selected independently for each of these three approximants, i.e. γ_{wPCA}, γ_{PU} and γ_{LME}. Additionally, a tolerance must be set for the Newton–Raphson solution of the dual optimization problem in Equation (7) required to evaluate the local *max-ent* shape functions.

In setting the tolerances, it is important to note that those involved in the neighbor search, see Equations (2,4), have a noticeable effect on the computational cost of the method. The values

Table I. Numerical parameters used to perform the computations.

Parameter	wPCA	PU	LME
γ	1.8	3.0, 6.0	0.6, 1.0, 1.4
TOL	10^{-8}	10^{-6}	10^{-10}
TOL_{NR}	—	—	10^{-12}

DOI: 10.1002/nme

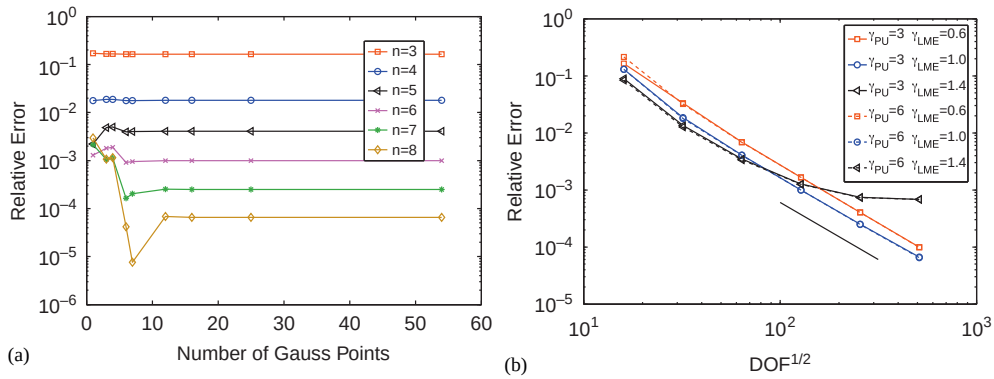


Figure 5. Relative error for the Willmore energy of a sphere. Error sensitivity with respect to (a) the number of quadrature points and (b) to the aspect-ratio parameter of the LME and PU shape functions. Also, a line of slope two for the visual inspection of (b) is shown.

chosen here provide good accuracy at a reasonable cost. The tolerance for the Newton–Raphson iterations influences very slightly the computational cost due to the fast quadratic convergence. As for the aspect ratio parameters, the calculations are robust for a wide range of values of γ_{wPCA}. The value selected in the Willmore energy calculations is shown in Table I, although a smaller value might be needed for data points affected by noise, or a higher value and more patch points for surfaces with sharp features. We analyze in more detail the sensitivity of the results with respect to the two remaining parameters, γ_{PU} and γ_{LME}.

We consider different levels of refinement of the point-set describing the sphere and of the set of patches. Specifically, we consider subdivisions of an octahedron following Loop's scheme, and relocate the resulting points on the unit sphere. We consider the levels of refinement $n=3,4,5,6,7,8$ with $N=2^{2(n+1)}+2$ points in P, and one level of refinement less for the patch points in Q, with $M=2^{2n}+2$ patches (see Figure 3). For this example, the refinement of the patches relative to the point-set P can be lowered by two or more levels without noticeable changes in the results.

Figure 5(a) shows the relative error e_{rel} for different levels of refinement and different number of cubature points. It can be observed that for coarse point-sets, the interpolation error dominates the quadrature error and the results are insensitive to the number of integration points. For finer point-sets, it can be observed that the accuracy is affected by the numerical quadrature, and up to 12 points per triangle are needed for the finest point-set, containing 262 146 points, in order to keep the quadrature error negligible relative to other errors.

Convergence plots in a log–log scale are shown in Figure 5(b) for different choices of the aspect ratio parameters γ_{LME} and γ_{PU}. The relative error in the Willmore energy converges with a slope of 2, i.e. $e_{\mathrm{rel}} \propto h^2$. The insensitivity of the results with respect to γ_{PU} is apparent. Nevertheless, this parameter has a significant impact on the computational cost. Indeed, low values cause much overlap and redundancy in the numerical quadrature, whereas higher values greatly reduce the number of quadrature points in each patch (see also Figure 3 for an illustration). However, excessively large values may result in poor overlapping and degradation of the accuracy. From our experience, the interval between 4 and 6 is a suitable range for γ_{PU}. As for γ_{LME}, we know that for very large values, the surface $\mathscr{M}_A$ will become polyhedral since the approximants will converge to the Delaunay shape functions. Therefore, for functionals depending on the curvature, lower values can be expected to produce accurate results. Indeed, we find numerically that the convergence is degraded for $\gamma_{\mathrm{LME}}=1.4$. In thin-shell calculations, lower values of γ_{LME} result in a denser structure of the stiffness matrix, hence in a higher computational cost. To conclude this section, we highlight the fact that accurate results and numerical evidence of convergence is provided throughout the paper for functionals depending on second-order spacial derivatives and their corresponding fourth-order partial differential equations, while using linearly consistent approximation schemes.

DOI: 10.1002/nme

3. THIN-SHELL MODEL

In this section, we review the mechanics of thin shells [11, 18], based on the geometrically exact formulation in [17, 19]. Here we restrict our attention to the linearized Kirchhoff–Love theory of shells, i.e. we neglect the shearing and stretching deformation normal to the shell mid-surface. In this theory, the shell director remains normal to the mid-surface during the deformation.

We follow the usual convention for Latin and Greek indices (i.e. $i=1,2,3$; $\alpha=1,2$) a comma denotes partial differentiation, subscripts refer to covariant components, and superscripts denote contravariant components.

3.1. Kinematics of the shell

We next describe the kinematics of a thin-shell body $\mathscr{S}\subset$ in three-space. We assume that this body can be described by the pair $(\boldsymbol{\varphi}, \boldsymbol{t})$, where the mapping $\boldsymbol{\varphi}$ defines the shell middle surface, Ω, and $\boldsymbol{t}$ is a field of unit vectors (a field of directors). We assume the thickness h of the shell to be uniform for simplicity. The thin shell body $\mathscr{S}$ is given by

$$\mathscr{S}=\left\{\boldsymbol{x}\in\mathbb{R}^3 | \boldsymbol{x}=\boldsymbol{\varphi}(\xi^\alpha)+\xi\boldsymbol{t}(\xi^\alpha), -\frac{h}{2}\leqslant\xi\leqslant\frac{h}{2}, (\xi^1,\xi^2)\in\mathscr{A}\right\}, \tag{11}$$

where $\mathscr{A}\subset\mathbb{R}^2$ is the parametric space for the middle surface, see Figure 6. Hence, we view a configuration $\boldsymbol{x}$ as a mapping from a parametric domain $\mathscr{A}\times[-h/2, h/2]$ into $\mathbb{R}^3$. The parametric domain is described by the coordinates $\{\xi^1, \xi^2, \xi^3\}$ (where we identify $\xi=\xi^3$), whose corresponding dual basis is $\{\boldsymbol{E}^i\}$. The area element of the middle surface can be computed as $d\Omega=\bar{j}d\xi^1 d\xi^2$, where $\bar{j}=\|\boldsymbol{\varphi}_{,1}\times\boldsymbol{\varphi}_{,2}\|$. The tangent map of a given configuration $T\boldsymbol{x}$ can be computed from the convective basis vectors $\boldsymbol{g}_i$ as

$$T\boldsymbol{x}=\frac{\partial\boldsymbol{x}}{\partial\xi^i}\otimes\boldsymbol{E}^i=\boldsymbol{g}_i\otimes\boldsymbol{E}^i,$$

with $\boldsymbol{g}_\alpha=\partial\boldsymbol{x}/\partial\xi^\alpha=\boldsymbol{\varphi}_{,\alpha}+\xi\boldsymbol{t}_{,\alpha}$ and $\boldsymbol{g}_3=\partial\boldsymbol{x}/\partial\xi=\boldsymbol{t}$. The covariant components of the metric tensor in convected coordinates are given by $g_{ij}=\boldsymbol{g}_i\cdot\boldsymbol{g}_j$.

Hereinafter, subscript 0 denotes quantities in the reference configuration, for instance $\boldsymbol{\varphi}_0$ is a point on the reference middle surface. A deformation mapping is a mapping from a reference body into $\mathbb{R}^3$, $\boldsymbol{x}\circ\boldsymbol{x}_0^{-1}$. Consequently, the deformation gradient is $\boldsymbol{F}=T\boldsymbol{x}(T\boldsymbol{x}_0)^{-1}$, and the Jacobian is $J=\det(\boldsymbol{F})=j/j_0$, where $j=\det(T\boldsymbol{x})=\boldsymbol{g}_3\cdot(\boldsymbol{g}_1\times\boldsymbol{g}_2)$.

The shell director in the reference configuration $\boldsymbol{t}_0$ coincides with the normal to the undeformed middle surface of the shell and hence has the properties

$$\boldsymbol{t}_0=\frac{\boldsymbol{\varphi}_{0,1}\times\boldsymbol{\varphi}_{0,2}}{\bar{j}_0},\quad \boldsymbol{\varphi}_{0,\alpha}\cdot\boldsymbol{t}_0=0,\quad |\boldsymbol{t}_0|=1,\quad \boldsymbol{t}_0\cdot\boldsymbol{t}_{0,\alpha}=0.$$

In general, the director in the deformed configuration of the shell, $\boldsymbol{t}$, is allowed to be an arbitrary vector field over $\Omega=\boldsymbol{x}(\mathscr{A}\times\{0\})$.

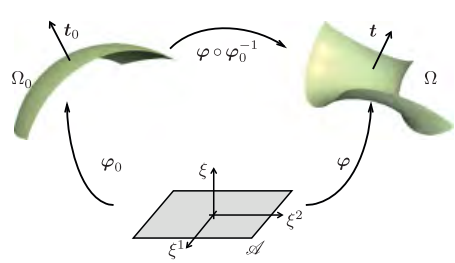


Figure 6. Schematic description of the reference, deformed and parametric configurations of the middle shell surface.

DOI: 10.1002/nme

The local shell deformations can be characterized by the Green–Lagrange strain tensor. Since the convected components of the metric tensor coincide with the components of $(T\boldsymbol{x})^{\mathrm{T}}T\boldsymbol{x}$ in the basis associated with $\{\xi^i\}$, the Green–Lagrange strain tensor can be expressed as the difference between the metric tensors on the deformed and undeformed configurations of the shell, i.e.

$$E_{ij}=\tfrac{1}{2}(g_{ij}-g_{0ij})=\tfrac{1}{2}(\boldsymbol{x}_{,i}\cdot\boldsymbol{x}_{,j}-\boldsymbol{x}_{0,i}\cdot\boldsymbol{x}_{0,j}).$$

Plugging the basic kinematic ansatz $\boldsymbol{x}=\boldsymbol{\varphi}(\xi^\alpha)+\xi\boldsymbol{t}(\xi^\alpha)$ into the above expression and grouping terms, we obtain

$$E_{ij}=\varepsilon_{ij}+\xi\rho_{ij}+(\xi)^2\vartheta_{ij}, \tag{12}$$

which admits the following interpretation in terms of the symmetric tensors ε_{ij}, ρ_{ij}, and ϑ_{ij}:

- The membrane strain tensor $\varepsilon_{\alpha\beta}=\frac{1}{2}(\boldsymbol{\varphi}_{,\alpha}\cdot\boldsymbol{\varphi}_{,\beta}-\boldsymbol{\varphi}_{0,\alpha}\cdot\boldsymbol{\varphi}_{,\beta})$, which lives on the middle surface, measures the in-plane deformation of the surface; the components $\varepsilon_{\alpha3}=\frac{1}{2}\boldsymbol{\varphi}_{,\alpha}\cdot\boldsymbol{t}$ measure the shearing of the director $\boldsymbol{t}_0$; and the component $\varepsilon_{33}=\frac{1}{2}(\boldsymbol{t}\cdot\boldsymbol{t}-1)$ measures the stretching of the director $\boldsymbol{t}_0$.
- The bending or change in curvature of the shell is measured by the tensor $\rho_{\alpha\beta}=\boldsymbol{\varphi}_{,\alpha}\cdot\boldsymbol{t}_{,\beta}-\boldsymbol{\varphi}_{0,\alpha}\cdot\boldsymbol{t}_{0,\beta}$, and $\rho_{\alpha3}=\frac{1}{2}\boldsymbol{t}_{,\alpha}\cdot\boldsymbol{t}$ measures the shearing originated from the director elongation; the in-plane tensor $\vartheta_{\alpha\beta}=\frac{1}{2}(\boldsymbol{t}_{,\alpha}\cdot\boldsymbol{t}_{,\beta}-\boldsymbol{t}_{0,\alpha}\cdot\boldsymbol{t}_{0,\beta})$ is exclusively related to changes of the middle surface directors. The rest of the components vanish, $\rho_{33}=\vartheta_{3i}=\vartheta_{i3}=0$.

3.2. Kirchhoff–Love hypothesis

In the remainder of this section, we restrict our attention to the Kirchhoff–Love theory of thin shells, i.e. we constrain the deformed director $\boldsymbol{t}$ to coincide with the unit normal of the deformed middle surface of the shell, i.e.

$$\boldsymbol{t}=\frac{\boldsymbol{\varphi}_{,1}\times\boldsymbol{\varphi}_{,2}}{\bar{j}},\quad \boldsymbol{\varphi}_{,\alpha}\cdot\boldsymbol{t}=0,\quad |\boldsymbol{t}|=1,\quad \boldsymbol{t}\cdot\boldsymbol{t}_{,\alpha}=0.$$

Consequently, the theory can be formulated exclusively in terms of the shell middle surface. We introduce its first and second fundamental forms expressed in convected components

$$a_{\alpha\beta}=\boldsymbol{\varphi}_{,\alpha}\cdot\boldsymbol{\varphi}_{,\beta},$$

$$\kappa_{\alpha\beta}=\boldsymbol{\varphi}_{,\alpha}\cdot\boldsymbol{t}_{,\beta}=-\boldsymbol{\varphi}_{,\alpha\beta}\cdot\boldsymbol{t}.$$

Here we have identified the director with the normal. With the Kirchhoff–Love hypothesis, the only remaining non-zero components of the Green–Lagrange strain tensor are

$$\begin{aligned}E_{\alpha\beta}&=\frac{1}{2}(a_{\alpha\beta}-a_{0\alpha\beta})+\xi(\kappa_{\alpha\beta}-\kappa_{0\alpha\beta})+\frac{(\xi)^2}{2}(\boldsymbol{t}_{,\alpha}\cdot\boldsymbol{t}_{,\beta}-\boldsymbol{t}_{0,\alpha}\cdot\boldsymbol{t}_{0,\beta})\\&=\varepsilon_{\alpha\beta}+\xi\rho_{\alpha\beta}+(\xi)^2\vartheta_{\alpha\beta}.\end{aligned} \tag{13}$$

3.3. Small displacements

As we say at the beginning of this section, we assume that the deformation field for the shell is restricted to account only for small displacement. Then, if the displacement vector of the middle surface of the shell is defined as $\boldsymbol{u}=\boldsymbol{\varphi}-\boldsymbol{\varphi}_0$, the linear membrane and bending strain measures can be derived from the kinematic variables up to first order in $\boldsymbol{u}$ in the form

$$\varepsilon_{\alpha\beta}=\tfrac{1}{2}(\boldsymbol{\varphi}_{0,\alpha}\cdot\boldsymbol{u}_{,\beta}+\boldsymbol{\varphi}_{0,\beta}\cdot\boldsymbol{u}_{,\alpha}),$$

$$\rho_{\alpha\beta}=-\boldsymbol{t}_0\cdot\boldsymbol{u}_{,\alpha\beta}-\boldsymbol{\varphi}_{0,\alpha\beta}\cdot\Delta\boldsymbol{t},$$

$$\vartheta_{\alpha\beta}=\tfrac{1}{2}(\boldsymbol{t}_{0,\alpha}\cdot\Delta\boldsymbol{t}_{,\beta}+\Delta\boldsymbol{t}_{,\alpha}\cdot\boldsymbol{t}_{0,\beta}),$$

DOI: 10.1002/nme

with

$$\boldsymbol{t}_{0,\alpha} = -\bar{j}_0^{-1}\{\boldsymbol{t}_0 \times [\boldsymbol{t}_0 \times (\boldsymbol{\varphi}_{0,1\alpha} \times \boldsymbol{\varphi}_{0,2} + \boldsymbol{\varphi}_{0,1} \times \boldsymbol{\varphi}_{0,2\alpha})]\},$$

$$\Delta\boldsymbol{t} = -\bar{j}_0^{-1}\{\boldsymbol{t}_0 \times [\boldsymbol{t}_0 \times (\boldsymbol{u}_{,1} \times \boldsymbol{\varphi}_{0,2} + \boldsymbol{\varphi}_{0,1} \times \boldsymbol{u}_{,2})]\},$$

where $\Delta\boldsymbol{t}$ is the approximation to $\boldsymbol{t}-\boldsymbol{t}_0$ up to first order in $\boldsymbol{u}$. We refer to the Appendix B for a detailed derivation of the linearized strains.

3.4. Equilibrium configuration of thin shells

The potential energy of an elastic shell body with internal energy density W can be expressed by the functional

$$\Pi[\boldsymbol{u}] = \int_{\mathscr{S}_0} W(\boldsymbol{u})\,\mathrm{d}V_0 + \Pi_{ext}[\boldsymbol{u}],$$

where Π_{ext} is the potential energy of the external loads. For concreteness, we consider an isotropic Kirchhoff–St. Venant elastic material, with an internal energy density expressed as [40]

$$W = \tfrac{1}{2} C^{ijkl} E_{ij} E_{kl},$$

where C^{ijkl} are the contravariant components of the elasticity tensor.

For thin-shell bodies, the Green–Lagrange tensor components are commonly retained up to first order in h, see Equation (13), and the effect of curvature on the configuration Jacobian away from the middle surface is neglected, that is $j_0/\bar{j}_0 = 1$ [17, 19]. Assuming that the elasticity tensor does not vary through-the-thickness, the internal energy density can be integrated through-the-thickness, resulting in an internal energy density per unit area

$$\mathscr{W} = \frac{1}{2}\int_{-h/2}^{h/2} C^{\alpha\beta\gamma\delta} E_{\alpha\beta} E_{\gamma\delta} \frac{j_0}{\bar{j}_0}\,\mathrm{d}\xi \simeq \frac{1}{2} C^{\alpha\beta\gamma\delta}\left(h\varepsilon_{\alpha\beta}\varepsilon_{\gamma\delta} + \frac{h^3}{12}\rho_{\alpha\beta}\rho_{\gamma\delta}\right),$$

with

$$C^{\alpha\beta\gamma\delta} = \frac{E}{(1-\nu^2)}\left[\nu a_0^{\alpha\beta} a_0^{\gamma\delta} + \frac{1}{2}(1-\nu)\left(a_0^{\alpha\gamma} a_0^{\beta\delta} + a_0^{\alpha\delta} a_0^{\beta\gamma}\right)\right],$$

where $a_0^{\alpha\gamma}(a_0)_{\gamma\beta} = \delta^{\alpha}_{\beta}$, E is Young's modulus, and ν the Poisson ratio. Thus, the internal potential energy can be written as an integral over the reference middle surface

$$\Pi_{int}[\boldsymbol{u}] = \int_{\Omega_0} \mathscr{W}(\boldsymbol{u})\,\mathrm{d}\Omega_0,$$

and the external potential becomes

$$\Pi_{ext}[\boldsymbol{u}] = -\int_{\Omega_0} \boldsymbol{q}\cdot\boldsymbol{u}\,\mathrm{d}\Omega_0 - \int_{\partial\Omega_0} \boldsymbol{h}\cdot\boldsymbol{u}\,\mathrm{d}\ell_0,$$

where $\boldsymbol{q}$ is the external body load per unit area, $\boldsymbol{h}$ the forces per unit length applied on the boundary of the middle surface, and $d\ell_0$ is the line element of the boundary of the middle surface.

The equilibrium displacement field is obtained by requiring stationarity of the total potential energy with respect to all admissible variations $\boldsymbol{\eta}$

$$\delta\Pi[\boldsymbol{u},\boldsymbol{\eta}] = \frac{\mathrm{d}}{\mathrm{d}\varepsilon}\Pi(\boldsymbol{u}+\varepsilon\boldsymbol{\eta})\bigg|_{\varepsilon=0} = 0.$$

Here, the admissible displacement fields and variations must be consistent with the essential boundary conditions. For thin shells, these can be in terms of the displacements, in $\partial\Omega_0^u$, or the rotations of the director about the tangent to the boundary, in $\partial\Omega_0^\theta$. Note that $\partial\Omega_0^u$ and $\partial\Omega_0^\theta$ can

DOI: 10.1002/nme

overlap. The above is a statement of the principle of virtual work, which can be expressed in terms of integrals over the parametric space $\mathscr{A}$ as follows:

$$\delta\Pi_{int}[\boldsymbol{u},\boldsymbol{\eta}] = \int_{\mathscr{A}} C^{\alpha\beta\gamma\delta}\left(h\varepsilon_{\gamma\delta}(\boldsymbol{u})\varepsilon_{\alpha\beta}(\boldsymbol{\eta}) + \frac{h^3}{12}\rho_{\gamma\delta}(\boldsymbol{u})\rho_{\alpha\beta}(\boldsymbol{\eta})\right)\bar{j}_0\,\mathrm{d}\xi^1\,\mathrm{d}\xi^2,$$

$$\delta\Pi_{ext}[\boldsymbol{\eta}] = -\int_{\mathscr{A}} \boldsymbol{q}\cdot\boldsymbol{\eta}\,\bar{j}_0\,\mathrm{d}\xi^1\,\mathrm{d}\xi^2 - \int_{\partial\mathscr{A}} \boldsymbol{h}\cdot\boldsymbol{\eta}\|\boldsymbol{\varphi}_{0,t}\|\,\mathrm{d}\ell_\xi,$$

where $\boldsymbol{\varphi}_{0,t}$ is the partial derivative of the middle surface mapping in the tangential direction to the boundary curve.

Following [17], we introduce the elastic constitutive relations between the shell stresses and the strains as

$$n^{\alpha\beta} = \frac{\partial\mathscr{W}}{\partial\varepsilon_{\alpha\beta}} = hC^{\alpha\beta\gamma\delta}\varepsilon_{\gamma\delta},$$

$$m^{\alpha\beta} = \frac{\partial\mathscr{W}}{\partial\rho_{\alpha\beta}} = \frac{h^3}{12}C^{\alpha\beta\gamma\delta}\rho_{\gamma\delta},$$

where $n^{\alpha\beta}$ is the effective membrane stress and $m^{\alpha\beta}$ is the effective bending stress, which can be interpreted as force and moment resultants. Further, by recourse to Voigt's notation, we obtain the following convenient expressions:

$$\boldsymbol{n} = \begin{pmatrix} n^{11} \\ n^{22} \\ n^{12} \end{pmatrix} = h\boldsymbol{C}\boldsymbol{\varepsilon}, \quad \boldsymbol{m} = \begin{pmatrix} m^{11} \\ m^{22} \\ m^{12} \end{pmatrix} = \frac{h^3}{12}\boldsymbol{C}\boldsymbol{\rho}, \quad \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ 2\varepsilon_{12} \end{pmatrix}, \quad \boldsymbol{\rho} = \begin{pmatrix} \rho_{11} \\ \rho_{22} \\ 2\rho_{12} \end{pmatrix},$$

where the matrix $\boldsymbol{C}$ is given by the expression

$$\boldsymbol{C} = \frac{E}{1-\nu^2}\begin{pmatrix} (a_0^{11})^2 & \nu a_0^{11}a_0^{22} + (1-\nu)(a_0^{12})^2 & a_0^{11}a_0^{12} \\ & (a_0^{22})^2 & a_0^{22}a_0^{12} \\ \text{symm} & & \frac{1}{2}[(1-\nu)a_0^{11}a_0^{22} - (1+\nu)(a_0^{12})^2] \end{pmatrix}.$$

Finally, with the above definitions, we can write the principle of virtual work as

$$0 = \delta\Pi[\boldsymbol{u},\delta\boldsymbol{u}] = \int_{\mathscr{A}} (\delta\boldsymbol{\varepsilon}\cdot\boldsymbol{n} + \delta\boldsymbol{\rho}\cdot\boldsymbol{m})\bar{j}_0\,\mathrm{d}\xi^1\,\mathrm{d}\xi^2 + \delta\Pi_{ext}[\delta\boldsymbol{u}]. \tag{14}$$

3.5. Galerkin discretization

We consider now the discrete equilibrium equations $\boldsymbol{KU} = \boldsymbol{f}$ for surfaces numerically represented with the procedure described before, in terms of a set of nodes $P = \{\boldsymbol{P}_a\}$, $a = 1, \ldots, N$, and a set of patches identified with the patch points $Q = \{\boldsymbol{Q}_A\}$, $A = 1, \ldots, M$. We consider a specific patch A. Let $\boldsymbol{\varphi}_{0A}$ be a configuration mapping for the middle surface for this patch, defined over the parametric space $\mathscr{A}_A$

$$\boldsymbol{\varphi}_{0A}(\boldsymbol{\xi}) = \sum_{a\in\hat{\mathscr{N}}^P_{\boldsymbol{Q}_A}} p_a(\boldsymbol{\xi})\boldsymbol{P}_a,$$

as described in Section 2.2. We represent the displacement field in a given patch A as

$$\boldsymbol{u}_A(\boldsymbol{\xi}) = \sum_{a\in\hat{\mathscr{N}}^P_{\boldsymbol{Q}_A}} p_a(\boldsymbol{\xi})\boldsymbol{u}_a.$$

DOI: 10.1002/nme

Virtual displacements are represented likewise. With the strategy presented in Section 2.3, we can split the integrals in the principle of virtual work into patch contributions, e.g.

$$\delta\Pi_{int}[\boldsymbol{u},\delta\boldsymbol{u}]=\sum_{A=1}^{M}\int_{\mathscr{A}_A}[(\delta\boldsymbol{\varepsilon}\cdot\boldsymbol{n}+\delta\boldsymbol{\rho}\cdot\boldsymbol{m})\bar{j}_0]_A(w_A^Q\circ\boldsymbol{\varphi}_0)\,\mathrm{d}\xi^1\,\mathrm{d}\xi^2. \tag{15}$$

Here, $[\cdot]_A$ means that the expression within the brackets is evaluated with the Ath patch approximation of the undeformed middle surface, the displacement field, and the virtual displacement field. A simple calculation yields the Galerkin stiffness matrix. The interaction between nodes a and b is given by

$$\boldsymbol{K}^{ab}=\sum_{A=1}^{M}\int_{\mathscr{A}_A}\left[\left(h\boldsymbol{M}^{a\mathrm{T}}\boldsymbol{C}\boldsymbol{M}^{b}+\frac{h^3}{12}\boldsymbol{B}^{a\mathrm{T}}\boldsymbol{C}\boldsymbol{B}^{b}\right)\bar{j}_0\right]_A(w_A^Q\circ\boldsymbol{\varphi}_0)\,\mathrm{d}\xi^1\,\mathrm{d}\xi^2,$$

where $\boldsymbol{M}^a$ and $\boldsymbol{B}^a$ are the membrane and bending strain–displacement matrices for the ath node. Note that $\boldsymbol{M}^a,\boldsymbol{B}^a\in\mathbb{R}^{3\times 3}$. See Appendix C for a detailed description. The force contribution of the ath node is

$$\boldsymbol{f}^a=\sum_{A=1}^{M}\int_{\mathscr{A}_A}[\boldsymbol{q}\;\; p_a\;\; \bar{j}_0]_A(w_A^Q\circ\boldsymbol{\varphi}_0)\,\mathrm{d}\xi^1\,\mathrm{d}\xi^2+\sum_{A=1}^{M}\int_{\partial\mathscr{A}_A}[\boldsymbol{h}p_a\|\boldsymbol{\varphi}_{0,t}\|]_A(w_A^Q\circ\boldsymbol{\varphi}_0)\,\mathrm{d}\ell_\xi.$$

The Dirichlet displacement and rotation boundary conditions are imposed with the Lagrange multipliers, as in [11], see Appendix D.

4. NUMERICAL EXAMPLES

We confront next the proposed method with a set of standard numerical benchmark tests. We consider a square plate loaded with a point force, and the shell obstacle course of Belytschko *et al.* [27] for the linear analysis of thin shells. The test problems in the shell obstacle course are Scordelis–Lo's roof loaded by self-weight, a pinched short cylinder, and a hemisphere loaded with two pairs of facing concentrated forces. At the end of this section, the flexibility of the proposed methodology to deal with the shells of complex topology and geometry is illustrated by two additional examples. Before presenting the numerical examples, a few remarks concerning the numerical implementation of the method are collected.

4.1. Numerical aspects

In all the calculations of this section, we fix $\gamma_{\mathrm{w}PCA}=1.8$ and $TOL_{\mathrm{wPCA}}=10^{-8}$ for the tangent plane weight functions, $\gamma_{\mathrm{PU}}=4.0$ and $TOL_{\mathrm{PU}}=10^{-6}$ for the PU shape functions, and $TOL_{\mathrm{LME}}=10^{-10}$ and $TOL_{\mathrm{NR}}=10^{-12}$ for the LME approximants. As for γ_{LME}, we discuss its influence with the first numerical example, and then fix it to $\gamma_{\mathrm{LME}}=0.8$, which provides accurate solutions at a moderate computational cost. We note that the present approach neither requires an external specification of the normals to the surface nor of the tangent vectors to the shell boundary [11].

4.1.1. Numerical quadrature. Numerical quadrature is one of the main outstanding issues in meshfree methods. Here, we follow the straightforward approach of building a Delaunay triangulation in the parametric domain of each patch $\mathscr{A}_A$, and placing quadrature points supported on this local planar triangulation. More advanced technologies exist that probably result in more efficient simulations [41]. In all the examples, we used a standard Gauss–Legendre cubature rule of 12 points (order 6) per triangle. In the boundary curves, a quadrature scheme of four Gauss–Legendre points per integration cell was used.

4.1.2. Boundary conditions. In principle, with the weak Kronecker-delta property of the local *max-ent* approximants [25] at the boundary, the displacement Dirichlet boundary conditions can be treated directly. However, we have noticed through numerical experiments that this results in

DOI: 10.1002/nme

locking, as the local *max-ent* approximants become flat (with zero curvature) near the boundary of the domain. This could be avoided using second-order *max-ent* approximants, a topic of current investigation.

We bypass this difficulty introducing ghost nodes, like in subdivision finite elements [18, 42]. This destroys the weak Kronecker-delta property on the boundary, but produces convergent approximations to shapes with non-vanishing curvature near the boundary. Consequently, we use the Lagrange multipliers both for the slope and for the displacement boundary conditions, see [11] and Appendix D.

4.1.3. Patch vs node points. A systematic study of the optimal relation between the number and location of the patch points relative to the node points has not been performed here. Related issues have been addressed in the area of computer graphics for surface reconstruction and rendering from unorganized set of points [6]. In the later work [43], a methodology to obtain the patch points as a down-sampling of the node points was presented in an MLS context. Through numerical tests, we have observed that the number of patch points needs to be large enough to capture the geometry of the surface. In our examples, we define refinement rules for the node sets, and choose a lower order of refinement for the patches than for the nodes. To give an idea of the extent of the patches and their ovelapping, for $\gamma_{\mathrm{PU}} \in [4, 6]$ and $TOL_{\mathrm{PU}} = 10^{-6}$, the support of the patch PU functions w_A^Q is of the order of $[1.52, 1.86]h_A$, where h_A is the characteristic spacing of the patch points.

4.2. Pinched plate

We consider a clamped square plate with a concentrated central force, see Figure 7(a). In this test, the bending behavior completely dominates the solution. The solution of the plate center deflection is $\delta_z = 0.00560FL^2/D$, where $D = Eh^3/(12(1-\nu^2))$ is the flexural rigidity of the plate [44]. The side length is $L = 10$, the thickness $h = 0.1$, the concentrated load applied at its center $F = 200$, Young's modulus $E = 10^6$, and the Poisson ratio $\nu = 0.3$. Material and geometrical parameters coincide with those used by Noels and Radovitzky [24], where a Discontinuous Galerkin method for thin shells is developed. Since the plate center deflection does not contain enough significant digits, we use a converged displacement obtained with a very fine discretization, $\delta_z = 0.0056122FL^2/D$, which is similar to the value reported in [24].

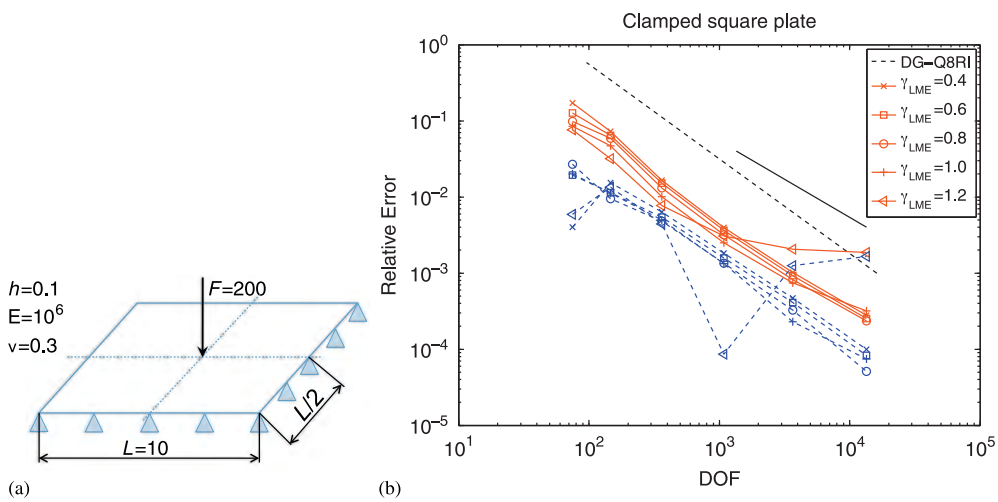


Figure 7. (a) Sketch for the pinched plate problem and (b) relative error for the center deflection. Results for the full model (—) and quarter model cases (- - -) are depicted. In (b) a line segment of slope two is shown to help in the visual inspection of the curves.

DOI: 10.1002/nme

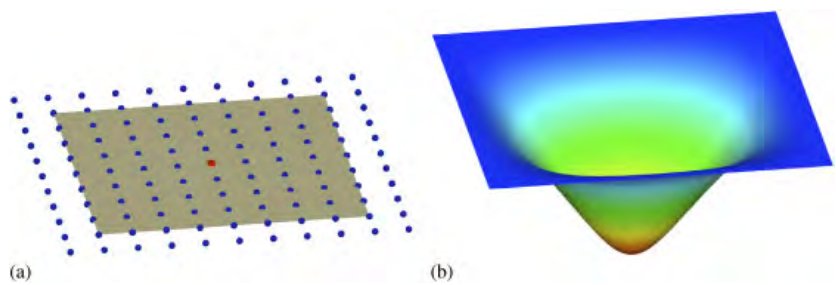


Figure 8. Full model of the clamped plate: (a) discretization and reference configuration and (b) computed deformed surface, $\gamma_{\mathrm{LME}}=0.8$. In the picture, the high smoothness of the deflected plate is noteworthy.

Owing to the symmetry of the problem, a quarter of the plate with appropriate symmetry boundary conditions can be considered. The convergence plots for the relative error of the center deflection for both full and quarter clamped plate models are illustrated in Figure 7(b). As expected, the models that take symmetry into account and consider one quarter of the plate have a denser node distribution for a given number of degrees of freedom, hence produce more accurate solutions. The results are compared with those obtained by using a DG method (values from [24]). The figure illustrates the high accuracy of the local *max-ent* approximations for $\gamma_{\mathrm{LME}}=0.4, 0.6, 0.8$. We observe a degradation of the convergence for $\gamma_{\mathrm{LME}}\geqslant 1.0$, similar to the behavior reported in Section 2.6. The superior accuracy of the proposed approach as compared with the DG method in this example is clear from the figure, with more than one order of magnitude of more accurate results. Note that these authors report the results for the quarter model; hence, their results need to be compared with the dashed lines in the figure. As the rate of convergence of the DG method is higher, this is not expectable for extremely fine discretizations. We note again that our method is based on smooth approximants that are only first-order consistent. Structured and unstructured sets of nodes have been used, with no significant difference in the result. Figure 8 illustrates the node set, with the ghost nodes, and the smooth deformation. In the remaining examples, by simplicity, a constant value of $\gamma_{\mathrm{LME}}=0.8$ has been selected.

4.3. Scordelis–Lo roof

In this example, a cylindrical roof is loaded by uniform gravity $g=90$, and supported by rigid diaphragms in the arched sides, $u_x=u_y=0$. Material and geometrical parameters are detailed in Figure 9(a). In the Scordelis–Lo's roof test, both the membrane and the flexural strain energies dominate the overall behavior of the model; hence, representation of inextensional modes is not crucial in this problem. Therefore, this test evaluates the inadequacies of the numerical method to compute membrane stresses, which severely could inhibit convergence.

The reference value for the maximum vertical displacement at the mid-side if the free edge reported in the literature presents a wide scatter, e.g. with a 3% difference between the converged value found here and that reported in [42]. For an overkill discretization we find $\delta_z=0.300575$, which is 0.6% lower than the value given in [27] of 0.3024. Figure 9(b) shows the convergence results obtained in both the full and quarter case, which exhibit a power-law asymptotic convergence behavior. Figure 10 shows typical node and patch point-sets, as well as the deformation for this example.

4.4. Pinched hemisphere

In this example, a hemispherical shell of radius $R=10$ and thickness $h=0.04$ is subjected to radial loads $F=2$ applied on two diametral directions, see Figure 11. This is a challenging test for assessing the method's ability to represent inextensional deformations under complex shell bending conditions involving curvature in two directions. The ability to bend without developing parasitic membrane strains is essential for good performance in this problem. Furthermore, the

DOI: 10.1002/nme

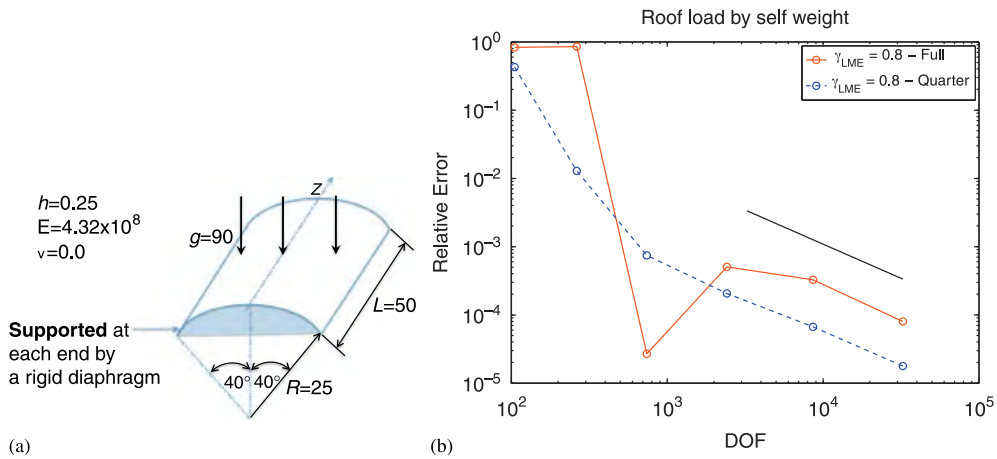


Figure 9. (a) Sketch for the Scordelis–Lo roof problem and (b) normalized deflection of the vertical displacement at the mid-side of the free edge, and line segment of slope two. Results are given for the full model (top —) and for the quarter model (bottom - - -).

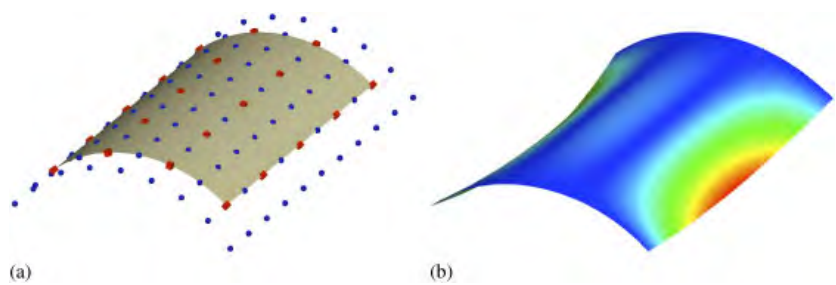


Figure 10. Full model of the Scordelis–Lo's roof: (a) discretization of both patches -◇- and nodes -○-, and the reference configuration and (b) computed deformed surface, $\gamma_{\mathrm{LME}} = 0.8$.

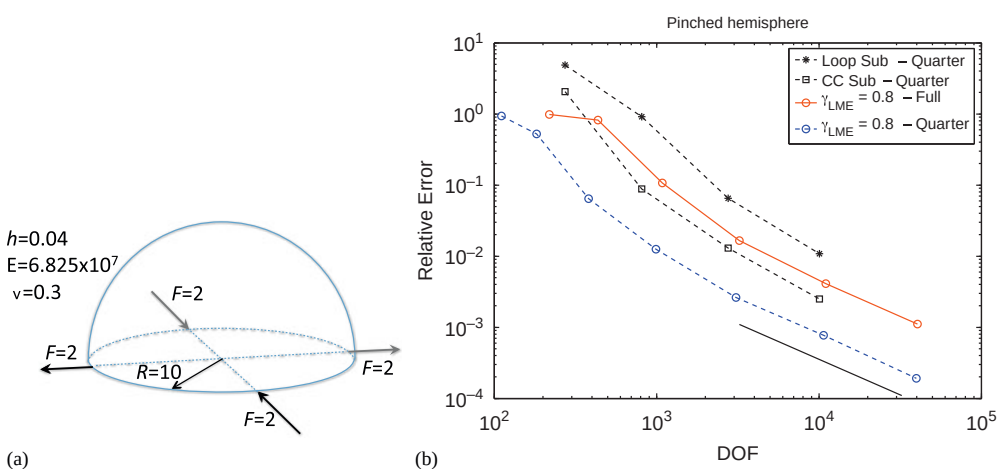


Figure 11. (a) Sketch for the pinched hemisphere problem and (b) pinched hemispherical shell problem. Results for the convergence of the normalized radial displacement are given the full model (top —) and for the quarter model (bottom - - -).

DOI: 10.1002/nme

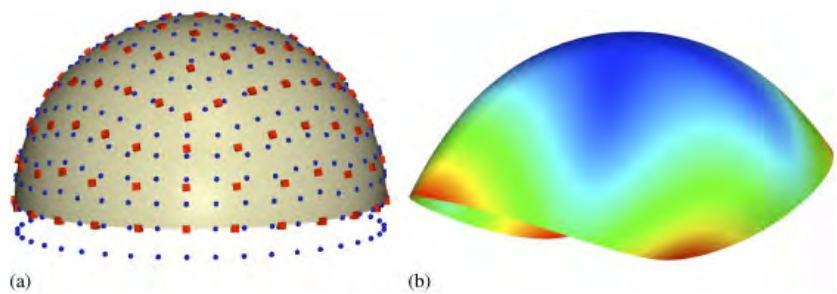


Figure 12. Full model of the pinched hemisphere: (a) discretization of both patches -◇- and nodes -○-, and reference configuration and (b) computed deformed surface, $\gamma_{\mathrm{LME}} = 0.8$.

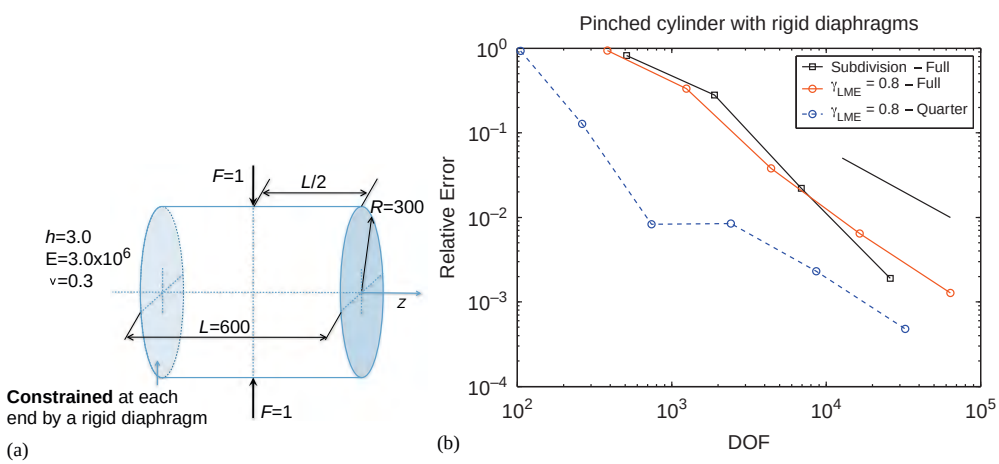


Figure 13. (a) Sketch for the pinched cylinder with rigid diaphragms and (b) pinched cylinder with rigid diaphragms problem. Convergence rates for the relative error of the deflection under the loads are given for the full model (top —) and for an eighth of the model (bottom - - -).

case of a hemispherical shell provides an example of a surface that presents necessarily irregular nodes, i.e. which cannot be meshed in a structured way without singularities. For instance, in the presence of such nodes, simple box-spline methods are not applicable, and a special treatment in subdivision finite element methods is required [18].

The convergence of the relative error for the radial displacement is shown in Figure 11, for both the full and quarter models. The displacements are normalized by a deflection value of $\delta_r = 0.092414$, which is in agreement with the lower bound value given by MacNeal and Harder [15, 27] and Belytschko *et al.* [27] , that is $\delta_r = 0.0924$. In this figure, we plot the convergence results reported in [42] for subdivision finite elements based on Loop's scheme (triangular elements) and on Catmull–Clark's scheme (quadrilateral elements). The superior accuracy of the proposed meshfree approach is clear from the figure, with over one order of magnitude better accuracy than the flexible approach based on Loop's scheme. The control and patch points, as well as the deformation, are shown in Figure 12.

4.5. Pinched cylinder

In the last example of the obstacle course, a short cylinder with rigid end diaphragms, $u_x = u_y = \theta_z = 0$, is subjected to compressive normal loads $F = 1$ applied along one diameter at the center of the cylinder, see Figure 13(a). The pinched-cylinder problem is a severe test for both inextensional bending modes and complex membrane states. The solution involves highly localized deformations

DOI: 10.1002/nme

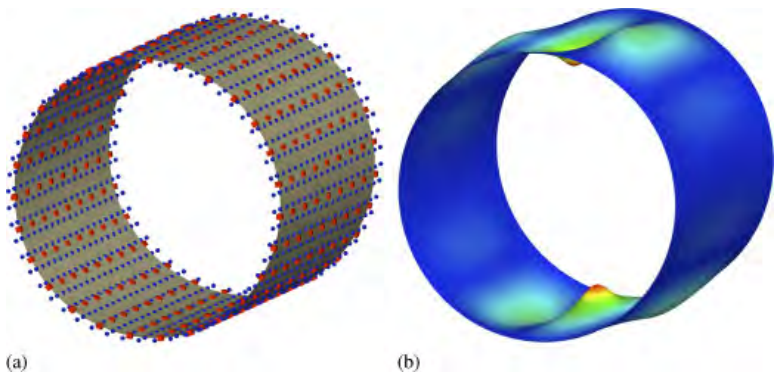


Figure 14. Full model of the pinched cylinder with rigid diaphragms: (a) discretization of both patches -◇- and nodes -○-, and reference configuration and (b) computed deformed surface, $\gamma_{\mathrm{LME}} = 0.8$.

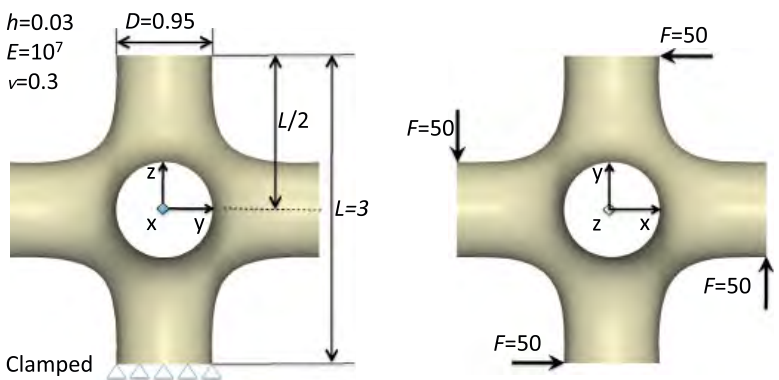


Figure 15. Sketch for the pinched pipes problem. Material and geometrical parameters are detailed. Also, the prescribed boundary conditions and the applied punctual forces are shown.

around the load, which preclude fast convergence. An exact solution for the displacement under the loads follows from a double Fourier series solution of Flugge's equations by Lindberg *et al.* [45], $\delta_r = 1.8248 \times 10^{-5} F$. Figure 13(b) presents the results for the full model case and for an eighth of the cylinder with the appropriate symmetry boundary conditions. In this case, the Catmull–Clark subdivision solutions reported in [42] is particularly accurate, slightly more than our approach for the finest meshes. The computed deformed limit surface is shown in Figure 14(b). The converged solution used as reference is $\delta_r = 1.7973 \times 10^{-5}$.

4.6. Pinched pipes

Here, we study the flexibility of the proposed approach in a shell with complex topology. Figure 15 shows the surface, consisting of a set of six connected open pipes subjected to four concentrated loads. In the figure both the location and the value of the loads are given, as well as the material and geometrical properties. The boundary curve of the bottom pipe is clamped. Figure 16 illustrates the discretization of the patch and the node points, which consist of two arrangements of 1628 and 6716 unstructured set of points, respectively. The original coarse point-set has been obtained from the MATLAB Central File exchange. Figure 17 shows the physical deformation obtained, which has been amplified by a factor of 2 to give a better idea of the resulting smooth field of displacements. We insist on the fact that this thin shell cannot be studied with the previous meshfree approaches because this surface does not admit a single parametric space.

DOI: 10.1002/nme

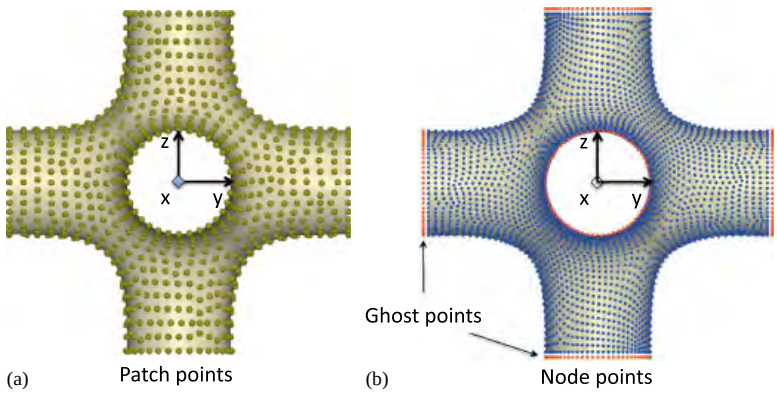


Figure 16. (a) Discretization used by computing the local parametric spaces, 1628 patch points and (b) node set distribution consisting of 6716 points, also the ghost nodes are detailed.

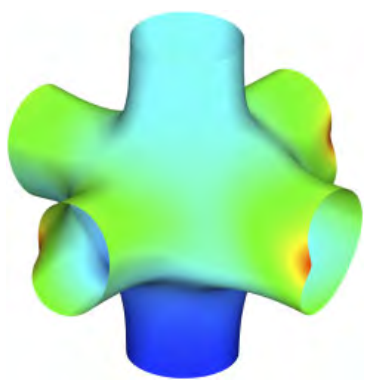


Figure 17. Physical deformation for the pinched pipes problem, $\gamma_{\mathrm{LME}} = 0.8$.

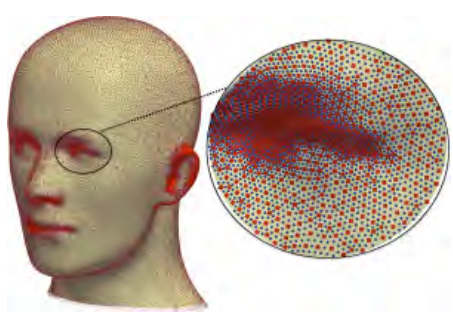


Figure 18. Discretization of the mannequin thin-shell problem, with 24 380 patch points and 97 675 node points.

4.7. Mannequin

This last example illustrates the ability of the method to deal with shells of complex geometry, defined by a set of points alone, without the need for a surface mesh. Of course, the quality of the node and patch sets needs to be sufficiently good, and adapted to complex geometric features, here the ears and the eyes. Figure 18 shows a mannequin elastic thin-shell under (a) a uniform gravity load of magnitude $g = 9.0$ and (b) two concentrated forces of value $F = 100$ at the crown of the head and at the nose. Material parameters have been selected as $E = 10^6$ and $v = 0.3$, whereas the

DOI: 10.1002/nme

Figure 19. Mannequin thin-shell deformations under the action of a uniform vertical load (left), and by applying two point forces (right).

thickness is $h = 0.08$. The boundary curve at the bottom is clamped. Figure 19 shows the smooth deformations obtained, showing strong effect of geometry on the deformation morphology.

5. CONCLUSIONS

We have presented a new methodology for processing d-dimensional point-set manifolds embedded in $\mathbb{R}^D$, which avoids a global parameterization or mesh. This approach relies on three ingredients: (1) the automatic detection of the local Euclidean structure of the manifold around a set of predefined patch points, i.e. the numerical tangent space to the manifold, (2) the local, smooth parameterization of the manifold around these patches using local *max-ent* approximants, and (3) a PU to split integrals into patch contributions. Each of these steps is general in dimension. We have applied the method to the Kirchhoff–Love thin-shell analysis. The performance of the method, assessed by the classical obstacle course, is excellent. It exhibits better accuracy for a given number of degrees of freedom than DG approaches to thin shells, and better or comparable results than subdivision finite elements. The proposed method significantly extends the applicability of meshfree methods to thin-shell analysis; in that it liberates such methods from the burden of requiring a single parametric space or imposing cumbersome patching conditions between meshfree macro-elements. This feature is illustrated by an example of a shell of complex topology. The method's applicability depends crucially on the quality of the sampling of the surface, specifically on the density of the sampling relative to the feature size. Such concepts have been formalized in the computer graphics literature [46].

Current research includes combining the proposed method with second-order *max-ent* approximants [26] to increase the order of convergence, improving the accuracy and efficiency of the numerical quadrature with stabilized nodal integration techniques [41]. More importantly, we are developing methods that dramatically reduce the number of patches (i.e. the number of local parameterizations and PU functions) relative to the number of nodes. With the current method, the density of patches is limited by the geometric features of the surface; in that the projections Π_A should not distort too much the node geometry. Finally, once the performance of the method has been assessed numerically, a mathematical analysis of the method shedding light on the rational choice of the numerical parameters would be highly desirable.

DOI: 10.1002/nme

APPENDIX A: DERIVATIVES OF THE SHAPE FUNCTIONS

We detail here the calculation of the derivatives of local *max-ent* approximants. We denote spatial gradients of scalar functions by ∇, whereas for vector-valued functions we denote by $D\boldsymbol{y}(\boldsymbol{\xi})$ the matrix of partial derivatives. The symbol ∂ denotes partial differentiation. The subindexes a, b, and c refer to nodes. Within the scope of the appendix, we define the following functions

$$f_a(\xi,\boldsymbol{\lambda}) = -\beta_a|\xi-\xi_a|^2+\boldsymbol{\lambda}\cdot(\xi-\xi_a), \tag{A1}$$

$$p_a(\xi,\boldsymbol{\lambda}) = \frac{\exp[\,f_a(\xi,\boldsymbol{\lambda})]}{\sum_b \exp[\,f_b(\xi,\boldsymbol{\lambda})]} = \frac{\exp[\,f_a(\xi,\boldsymbol{\lambda})]}{Z(\xi,\boldsymbol{\lambda})}, \tag{A2}$$

$$\boldsymbol{r}(\xi,\boldsymbol{\lambda}) = \sum_a p_a(\xi,\boldsymbol{\lambda})(\xi-\xi_a), \tag{A3}$$

$$\boldsymbol{J}(\xi,\boldsymbol{\lambda}) = \frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\lambda}} = \sum_a p_a(\xi,\boldsymbol{\lambda})(\xi-\xi_a)\otimes(\xi-\xi_a) - \boldsymbol{r}(\xi,\boldsymbol{\lambda})\otimes\boldsymbol{r}(\xi,\boldsymbol{\lambda}). \tag{A4}$$

The dependence on the evaluation point ξ and on the Lagrange multiplier λ is dropped for notational simplicity. The symbol $*$ is used to denote that a function is evaluated in $\boldsymbol{\lambda}^*(\xi) = \arg\min_{\boldsymbol{\lambda}\in\mathbb{R}^d} \ln Z(\xi,\boldsymbol{\lambda})$. This introduces explicit and implicit dependences on ξ in all functions with *. Note that what has been denoted by p_a in the remainder of the paper is denoted by p_a^* in the appendix. No implied sum is assumed for repeated node indices.

The first spatial derivative of the shape functions will be referred as ∇p_a^*. It is readily verified [25] that

$$\nabla p_a^* = p_a^*\left(\nabla f_a^* - \sum_c p_c^* \nabla f_c^*\right). \tag{A5}$$

Applying the chain rule, we have

$$\nabla f_a^* = \left(\frac{\partial f_a}{\partial \xi}\right)^* + D\boldsymbol{\lambda}^*\left(\frac{\partial f_a}{\partial \boldsymbol{\lambda}}\right)^*, \tag{A6}$$

where

$$\left(\frac{\partial f_a}{\partial \xi}\right)^* = -2\beta_a(\xi-\xi_a)+\boldsymbol{\lambda}^*, \quad \left(\frac{\partial f_a}{\partial \boldsymbol{\lambda}}\right)^* = (\xi-\xi_a).$$

The only term that is not available explicitly in Equation (A6) is $D\boldsymbol{\lambda}^*$. In order to compute it we note that, since $\boldsymbol{r}^*$ is identically zero,

$$0 = D\boldsymbol{r}^* = \left(\frac{\partial \boldsymbol{r}}{\partial \xi}\right)^* + D\boldsymbol{\lambda}^*\left(\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\lambda}}\right)^*,$$

where

$$\left(\frac{\partial \boldsymbol{r}}{\partial \boldsymbol{\lambda}}\right)^* = \boldsymbol{J}^*, \quad \left(\frac{\partial \boldsymbol{r}}{\partial \xi}\right)^* = -\boldsymbol{J}_\beta + \boldsymbol{I}, \quad \boldsymbol{J}_\beta = 2\sum_a \beta_a p_a^*(\xi-\xi_a)\otimes(\xi-\xi_a).$$

It follows that

$$D\boldsymbol{\lambda}^* = (\boldsymbol{J}_\beta - \boldsymbol{I})(\boldsymbol{J}^*)^{-1}.$$

Rearranging terms, we finally obtain the spatial gradients of the shape functions as

$$\nabla p_a^* = p_a^*\left(\boldsymbol{r}_\beta - \boldsymbol{M}_a(\xi-\xi_a)\right),$$

where

$$\boldsymbol{r}_\beta = 2\sum_a \beta_a p_a^*(\xi-\xi_a), \qquad \boldsymbol{M}_a = 2\beta_a \boldsymbol{I} - D\boldsymbol{\lambda}^*.$$

DOI: 10.1002/nme

The second spatial derivative of the shape functions, i.e. their Hessian, will be referred as $(Hp_a)^*$. We calculate the derivative of ∇p_a as

$$(Hp_a)^* = \underbrace{\nabla p_a^* \otimes \left(\nabla f_a^* - \sum_b p_b^* \nabla f_b^* \right)}_{A} + \underbrace{p_a^* \left((D\nabla f_a)^* - \sum_b p_b^* (D\nabla f_b)^* \right)}_{B} - \underbrace{p_a^* \sum_b \nabla p_b^* \otimes \nabla f_b^*}_{C},$$

where

$$A = p_a^* \left[\boldsymbol{r}_\beta - \boldsymbol{M}_a(\xi - \xi_a) \right] \otimes \left[\boldsymbol{r}_\beta - \boldsymbol{M}_a(\xi - \xi_a) \right],$$

$$B = 2p_a^* \left(\sum_b \beta_b p_b^* - \beta_a \right) \boldsymbol{I} + p_a^* D^2 \boldsymbol{\lambda}^*(\xi - \xi_a),$$

$$C = p_a^* \boldsymbol{r}_\beta \otimes \boldsymbol{r}_\beta - p_a^* \sum_b p_b^* \boldsymbol{M}_b(\xi - \xi_b) \otimes \boldsymbol{M}_b(\xi - \xi_b).$$

The term $D^2\boldsymbol{\lambda}^*$ is computed by using again the fact that $\boldsymbol{r}^*$ is identically zero, which also implies that $D^2\boldsymbol{r}^* = 0$. Lengthy but simple calculations lead to

$$D^2\boldsymbol{\lambda}^*(\xi - \xi_a) = \boldsymbol{r}_\beta \otimes \boldsymbol{j}_a + \boldsymbol{j}_a \otimes \boldsymbol{r}_\beta + (\boldsymbol{r}_\beta \cdot \boldsymbol{j}_a)\boldsymbol{I} - \sum_b p_b^* \Delta_{ab} \boldsymbol{M}_b(\xi - \xi_b) \otimes \boldsymbol{M}_b(\xi - \xi_b),$$

where

$$\Delta_{ab} = (\xi - \xi_b) \cdot (\boldsymbol{J}^*)^{-1}(\xi - \xi_a), \qquad \boldsymbol{j}_a = (\boldsymbol{J}^*)^{-1}(\xi - \xi_a).$$

Finally, the second spatial derivative of the shape functions can be written as

$$\begin{aligned}(Hp_a)^* = {} & p_a^* [\boldsymbol{r}_\beta - \boldsymbol{M}_a(\xi - \xi_a)] \otimes [\boldsymbol{r}_\beta - \boldsymbol{M}_a(\xi - \xi_a)] + 2p_a^* \left(\sum_b \beta_b p_b^* - \beta_a \right) \boldsymbol{I} \\ & + p_a^* [\boldsymbol{r}_\beta \otimes \boldsymbol{r}_\beta + \boldsymbol{r}_\beta \otimes \boldsymbol{j}_a + \boldsymbol{j}_a \otimes \boldsymbol{r}_\beta + (\boldsymbol{r}_\beta \cdot \boldsymbol{j}_a)\boldsymbol{I}] \\ & - p_a^* \sum_b p_b^* (1 + \Delta_{ab}) \boldsymbol{M}_b(\xi - \xi_b) \otimes \boldsymbol{M}_b(\xi - \xi_b). \end{aligned} \tag{A7}$$

APPENDIX B: STRAIN TENSORS FOR LINEARIZED KINEMATICS

In this appendix, we derive the bending strain tensor components for linearized kinematics. From Section 3, with the Kirchhoff–Love hypothesis and defining the displacement vector $\boldsymbol{u} = \boldsymbol{\varphi} - \boldsymbol{\varphi}_0$, the bending strains in Equation (12) become

$$\rho_{\alpha\beta} = \boldsymbol{\varphi}_{0,\alpha\beta} \cdot \boldsymbol{t}_0 - (\boldsymbol{\varphi}_{0,\alpha\beta} + \boldsymbol{u}_{,\alpha\beta}) \cdot \boldsymbol{t}, \tag{B1}$$

where

$$\boldsymbol{t} = \bar{j}^{-1}(\boldsymbol{\varphi}_{0,1} \times \boldsymbol{\varphi}_{0,2} + \boldsymbol{u}_{,1} \times \boldsymbol{\varphi}_{0,2} + \boldsymbol{\varphi}_{0,1} \times \boldsymbol{u}_{,2} + \boldsymbol{u}_{,1} \times \boldsymbol{u}_{,2}).$$

Neglecting higher-order terms in $\boldsymbol{u}$, $\bar{j}^{-1}$ expressed in the reference configuration takes the form

$$\bar{j}^{-1} \approx \bar{j}_0^{-1} - \bar{j}_0^{-2} \boldsymbol{t}_0 \cdot (\boldsymbol{u}_{,1} \times \boldsymbol{\varphi}_{0,2} + \boldsymbol{\varphi}_{0,1} \times \boldsymbol{u}_{,2}),$$

which allows us to calculate the normal director increment $\Delta\boldsymbol{t} = \boldsymbol{t} - \boldsymbol{t}_0$ as

$$\Delta\boldsymbol{t} \approx \bar{j}_0^{-1}(\boldsymbol{u}_{,1} \times \boldsymbol{\varphi}_{0,2} + \boldsymbol{\varphi}_{0,1} \times \boldsymbol{u}_{,2}) - \bar{j}_0^{-1}[\boldsymbol{t}_0 \cdot (\boldsymbol{u}_{,1} \times \boldsymbol{\varphi}_{0,2} + \boldsymbol{\varphi}_{0,1} \times \boldsymbol{u}_{,2})]\boldsymbol{t}_0.$$

This expression can be written more compactly after introducing $\boldsymbol{v} = \boldsymbol{u}_{,1} \times \boldsymbol{\varphi}_{0,2} + \boldsymbol{\varphi}_{0,1} \times \boldsymbol{u}_{,2}$ and the identity $\boldsymbol{a} \times (\boldsymbol{b} \times \boldsymbol{c}) = \boldsymbol{b}(\boldsymbol{a} \cdot \boldsymbol{c}) - \boldsymbol{c}(\boldsymbol{a} \cdot \boldsymbol{b})$:

$$\Delta\boldsymbol{t} = \bar{j}_0^{-1}[\boldsymbol{v}(\boldsymbol{t}_0 \cdot \boldsymbol{t}_0) - (\boldsymbol{t}_0 \cdot \boldsymbol{v})\boldsymbol{t}_0] = \bar{j}_0^{-1}[\boldsymbol{t}_0 \times (\boldsymbol{v} \times \boldsymbol{t}_0)].$$

DOI: 10.1002/nme

Replacing $\boldsymbol{t}$ by $\boldsymbol{t}_0+\Delta\boldsymbol{t}$ in Equation (B1), rearranging terms, applying the identities $\boldsymbol{a}\cdot(\boldsymbol{b}\times\boldsymbol{c})=\boldsymbol{c}\cdot(\boldsymbol{a}\times\boldsymbol{b})=\boldsymbol{b}\cdot(\boldsymbol{c}\times\boldsymbol{a})$ and $\boldsymbol{a}\times\boldsymbol{b}=-\boldsymbol{b}\times\boldsymbol{a}$, and neglecting higher-order terms in $\boldsymbol{u}$, the bending strains can be expressed as

$$\begin{aligned}\rho_{\alpha\beta}=&-\boldsymbol{t}_0\cdot\boldsymbol{u}_{,\alpha\beta}+\bar{j}_0^{-1}[(\boldsymbol{\varphi}_{0,\alpha\beta}\times\boldsymbol{\varphi}_{0,2})\cdot\boldsymbol{u}_{,1}+(\boldsymbol{\varphi}_{0,1}\times\boldsymbol{\varphi}_{0,\alpha\beta})\cdot\boldsymbol{u}_{,2}]\\&+\bar{j}_0^{-1}(\boldsymbol{t}_0\cdot\boldsymbol{\varphi}_{0,\alpha\beta})[(\boldsymbol{\varphi}_{0,2}\times\boldsymbol{t}_0)\cdot\boldsymbol{u}_{,1}+(\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,1})\cdot\boldsymbol{u}_{,2}].\end{aligned}\tag{B2}$$

The derivatives of the normal vector $\boldsymbol{t}_0$ are computed as

$$\boldsymbol{t}_{0,\alpha}=\bar{j}_0^{-1}(\boldsymbol{\varphi}_{,1\alpha}\times\boldsymbol{\varphi}_{0,2}+\boldsymbol{\varphi}_{0,1}\times\boldsymbol{\varphi}_{,2\alpha})-\bar{j}_0^{-1}[\boldsymbol{t}_0\cdot(\boldsymbol{\varphi}_{,1\alpha}\times\boldsymbol{\varphi}_{0,2}+\boldsymbol{\varphi}_{0,1}\times\boldsymbol{\varphi}_{,2\alpha})]\boldsymbol{t}_0,$$

which can be compactly re-written by applying the previous procedure for $\Delta\boldsymbol{t}$:

$$\boldsymbol{t}_{0,\alpha}=-\bar{j}_0^{-1}\{\boldsymbol{t}_0\times[\boldsymbol{t}_0\times(\boldsymbol{\varphi}_{0,1\alpha}\times\boldsymbol{\varphi}_{0,2}+\boldsymbol{\varphi}_{0,1}\times\boldsymbol{\varphi}_{0,2\alpha})]\}.$$

APPENDIX C: MEMBRANE AND BENDING STRAIN–DISPLACEMENT MATRICES

In this appendix we describe the membrane and bending strain–displacement matrices, which are needed for the computation of the stiffness matrix $\boldsymbol{K}$ (Section 3). We consider a specific patch A. Let $\boldsymbol{\varphi}_{0A}$, $\boldsymbol{\varphi}_0$ to keep the notation light, be a configuration mapping for the middle surface for this patch, defined over the parametric space $\mathscr{A}_A$

$$\boldsymbol{\varphi}_0(\boldsymbol{\xi})=\sum_{a\in\hat{\mathscr{N}}^P_{Q_A}}p_a(\boldsymbol{\xi})\boldsymbol{P}_a,$$

with derivatives

$$\boldsymbol{\varphi}_{0,\alpha}(\boldsymbol{\xi})=\sum_{a\in\hat{\mathscr{N}}^P_{Q_A}}p_{a,\alpha}(\boldsymbol{\xi})\boldsymbol{P}_a,\qquad\boldsymbol{\varphi}_{0,\alpha\beta}(\boldsymbol{\xi})=\sum_{a\in\hat{\mathscr{N}}^P_{Q_A}}p_{a,\alpha\beta}(\boldsymbol{\xi})\boldsymbol{P}_a.$$

The membrane and bending strain–displacement matrices for the ath nodal point, $\boldsymbol{M}^a$ and $\boldsymbol{B}^a$ respectively, take the form

$$M^a_{ij}=\boldsymbol{M}^a_i\cdot\boldsymbol{e}_j\quad\text{and}\quad B^a_{ij}=\boldsymbol{B}^a_i\cdot\boldsymbol{e}_j,$$

where

$$\begin{aligned}\boldsymbol{M}^a_\alpha&=p_{a,\alpha}\boldsymbol{\varphi}_{0,\alpha},\\\boldsymbol{M}^a_3&=p_{a,2}\boldsymbol{\varphi}_{0,1}+p_{a,1}\boldsymbol{\varphi}_{0,2},\\\boldsymbol{B}^a_\alpha&=-p_{a,\alpha\alpha}\boldsymbol{t}_0+\bar{j}_0^{-1}[(\boldsymbol{\varphi}_{0,\alpha\alpha}\times\boldsymbol{\varphi}_{0,2})p_{a,1}+(\boldsymbol{\varphi}_{0,1}\times\boldsymbol{\varphi}_{0,\alpha\alpha})p_{a,2}]\\&\quad+\bar{j}_0^{-1}(\boldsymbol{t}_0\cdot\boldsymbol{\varphi}_{0,\alpha\alpha})[(\boldsymbol{\varphi}_{0,2}\times\boldsymbol{t}_0)p_{a,1}+(\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,1})p_{a,2}],\\\boldsymbol{B}^a_3&=-2p_{a,12}\boldsymbol{t}_0+2\bar{j}_0^{-1}[(\boldsymbol{\varphi}_{0,12}\times\boldsymbol{\varphi}_{0,2})p_{a,1}+(\boldsymbol{\varphi}_{0,1}\times\boldsymbol{\varphi}_{0,12})]p_{a,2}\\&\quad+2\bar{j}_0^{-1}(\boldsymbol{t}_0\cdot\boldsymbol{\varphi}_{0,12})[(\boldsymbol{\varphi}_{0,2}\times\boldsymbol{t}_0)p_{a,1}+(\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,1})p_{a,2}],\end{aligned}$$

and $\boldsymbol{e}_j$ represent the canonical basis vectors of $\mathbb{R}^3$. Note that repeated indices in the expressions for $\boldsymbol{M}^a_\alpha$ and $\boldsymbol{B}^a_\alpha$ do not imply summation.

DOI: 10.1002/nme

APPENDIX D: ESSENTIAL BOUNDARY CONDITIONS

We describe here the imposition of the essential boundary conditions, both for the displacements and the rotations. We describe the variational formulation with the Lagrange multipliers (a slight variation of [11]), as well as the discretization.

Let us consider the integral of a function f over the boundary surface $\partial\mathscr{S}_0$ of a thin shell object $\mathscr{S}_0$. In fact, $\partial\mathscr{S}_0$ is the only thin part of the boundary of $\mathscr{S}_0$, i.e. it excludes the surfaces parallel to the middle surface. Assuming that the function does not change through-the-thickness, we have

$$\int_{\partial\mathscr{S}_0} f\,\mathrm{d}S_0 = \int_{\partial\Omega_0} f\left(\int_{-h/2}^{h/2} \frac{\left\|\dfrac{\partial \boldsymbol{x}_0}{\partial\xi}\times\dfrac{\partial \boldsymbol{x}_0}{\partial t}\right\|}{\|\boldsymbol{\varphi}_{0,t}\|}\,\mathrm{d}\xi\right)\mathrm{d}\ell_0,$$

where t is a tangent coordinate along the boundary curve $\partial\mathscr{A}$. By introducing $\partial\boldsymbol{x}_0/\partial t=\boldsymbol{\varphi}_{0,t}+\xi\boldsymbol{t}_{0,t}$, we obtain

$$\int_{-h/2}^{h/2} \frac{\left\|\boldsymbol{t}_0\times\dfrac{\partial \boldsymbol{x}_0}{\partial t}\right\|}{\|\boldsymbol{\varphi}_{0,t}\|}\,\mathrm{d}\xi = h\frac{\|\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,t}\|}{\|\boldsymbol{\varphi}_{0,t}\|}.$$

With the previous expressions and the PU, the integral of a function f on the boundary surface $\partial\mathscr{S}_0$ becomes

$$\begin{aligned}\int_{\partial\mathscr{S}_0} f\,\mathrm{d}S_0 &= \int_{\partial\Omega_0} hf\frac{\|\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,t}\|}{\|\boldsymbol{\varphi}_{0,t}\|}\,\mathrm{d}\ell_0\\ &= \sum_{A=1}^{M}\int_{\partial\mathscr{A}_A}[h(f\circ\boldsymbol{\varphi}_0)\|\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,t}\|]_A(w_A^Q\circ\boldsymbol{\varphi}_0)\,\mathrm{d}\ell_\xi.\end{aligned}$$

Here subindex A means that the expression between the brackets is computed with the local parameterization of the Ath patch.

D.1. Displacement constraints

Let $\boldsymbol{\lambda}_u$ be the Lagrange multipliers associated with the displacement constraints $\boldsymbol{u}=\bar{\boldsymbol{u}}$ on $\partial\Omega_0^u$; the additional terms for Equation (14) are

$$-\int_{\partial\Omega_0^u} h[(\boldsymbol{u}-\bar{\boldsymbol{u}})\cdot\delta\boldsymbol{\lambda}_u+\boldsymbol{\lambda}_u\cdot\delta\boldsymbol{u}]\frac{\|\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,t}\|}{\|\boldsymbol{\varphi}_{0,t}\|}\,\mathrm{d}\ell_0.$$

With the PU, we obtain

$$-\sum_{A=1}^{M}\int_{\partial\Omega_{0A}^u} h[(\boldsymbol{u}-\bar{\boldsymbol{u}})\cdot\delta\boldsymbol{\lambda}_u+\boldsymbol{\lambda}_u\cdot\delta\boldsymbol{u}]\frac{\|\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,t}\|}{\|\boldsymbol{\varphi}_{0,t}\|}w_A^Q\,\mathrm{d}\ell_0,$$

where $\partial\Omega_{0A}^u=\partial\Omega_0^u\cap supp(w_A^Q)$. Then, the final expression in the parametric space for the displacement constraints is

$$-\sum_{A=1}^{M}\int_{\partial\mathscr{A}_A^u}\{h[(\boldsymbol{u}-\bar{\boldsymbol{u}})\cdot\delta\boldsymbol{\lambda}_u+\boldsymbol{\lambda}_u\cdot\delta\boldsymbol{u}]\|\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,t}\|\}_A(w_A^Q\circ\boldsymbol{\varphi}_0)\,\mathrm{d}\ell_\xi.$$

D.2. Rotation constraints

We express a unit vector tangent to the boundary curve of the middle surface Ω_0 as $\boldsymbol{\tau}_0=\boldsymbol{\varphi}_{0,t}/\|\boldsymbol{\varphi}_{0,t}\|$, which satisfies $\boldsymbol{t}_0\cdot\boldsymbol{\tau}_0=0$. The small rotation vector $\boldsymbol{\theta}$ measures the infinitesimal rotation of the director field, i.e. it is given by the relation [11]

$$\boldsymbol{\theta}\times\boldsymbol{t}_0\approx\Delta\boldsymbol{t}=\boldsymbol{t}-\boldsymbol{t}_0,$$

DOI: 10.1002/nme

where the approximate sign means here up to first order in the displacements. Recalling the linearized expression for $\Delta \boldsymbol{t}$ of Appendix B

$$\Delta \boldsymbol{t} \approx -\bar{j}_0^{-1}\{[(\boldsymbol{u}_{,1}\times\boldsymbol{\varphi}_{0,2}+\boldsymbol{\varphi}_{0,1}\times\boldsymbol{u}_{,2})\times\boldsymbol{t}_0]\times\boldsymbol{t}_0\},$$

and rearranging the term between brackets as

$$(\boldsymbol{u}_{,1}\times\boldsymbol{\varphi}_{0,2}+\boldsymbol{\varphi}_{0,1}\times\boldsymbol{u}_{,2})\times\boldsymbol{t}_0=(\boldsymbol{u}_{,1}\cdot\boldsymbol{t}_0)\boldsymbol{\varphi}_{0,2}-(\boldsymbol{u}_{,2}\cdot\boldsymbol{t}_0)\boldsymbol{\varphi}_{0,1},$$

we finally obtain

$$\boldsymbol{\theta}=\bar{j}_0^{-1}[(\boldsymbol{u}_{,2}\cdot\boldsymbol{t}_0)\boldsymbol{\varphi}_{0,1}-(\boldsymbol{u}_{,1}\cdot\boldsymbol{t}_0)\boldsymbol{\varphi}_{0,2}]. \tag{D1}$$

The rotation of the shell boundary about the tangential direction is computed as $\theta_\tau=\boldsymbol{\theta}\cdot\boldsymbol{\tau}_0$. The rotation boundary conditions take the form $\theta_\tau=\bar{\theta}_\tau$ on $\partial\Omega_0^\theta$. Denoting with λ_θ the Lagrange multiplier associated with this constraint, the extra terms for the weak form of Equation (14) are

$$-\int_{\partial\Omega_0^\theta} h[(\theta_\tau-\bar{\theta}_\tau)\delta\lambda_\theta+\lambda_\theta\delta\theta_\tau]\,\mathrm{d}\ell_0.$$

With the PU, these terms become

$$-\sum_{A=1}^{M}\int_{\partial\mathscr{A}_A^\theta}\{h[(\theta_\tau-\bar{\theta}_\tau)\delta\lambda_\theta+\lambda_\theta\delta\theta_\tau]\|\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,t}\|\}_A(w_A^Q\circ\boldsymbol{\varphi}_0)\,\mathrm{d}\ell_\xi.$$

D.3. Numerical imposition of the constraints

The discretization of the displacement and rotational Lagrange multipliers is performed as follows:

$$\boldsymbol{\lambda}_u=\sum_a N_a\boldsymbol{\Lambda}_{ua},\quad \lambda_\tau=\sum_a N_a\Lambda_{\theta a},$$

where N_a are the basis functions that define the space of Lagrange multipliers. Here, we use the piecewise linear hat shape functions associated with the boundary nodes. The rotational constraints can be written in terms of the displacement nodal values as

$$\theta_\tau=\bar{j}_0^{-1}\sum_{b=1}^{N}[p_{b,2}(\boldsymbol{\varphi}_{0,1}\cdot\boldsymbol{\tau}_0)-p_{b,1}(\boldsymbol{\varphi}_{0,2}\cdot\boldsymbol{\tau}_0)](\boldsymbol{u}_b\cdot\boldsymbol{t}_0).$$

The solution is computed by solving the following linear system of equations:

$$\begin{pmatrix}\boldsymbol{K} & \boldsymbol{C}_u^{\mathrm{T}} & \boldsymbol{C}_\theta^{\mathrm{T}}\\ \boldsymbol{C}_u & \boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{C}_\theta & \boldsymbol{0} & \boldsymbol{0}\end{pmatrix}\begin{pmatrix}\boldsymbol{U}\\ \boldsymbol{\Lambda}_u\\ \boldsymbol{\Lambda}_\theta\end{pmatrix}=\begin{pmatrix}\boldsymbol{f}\\ \boldsymbol{g}_u\\ \boldsymbol{g}_\theta\end{pmatrix},$$

For concreteness, only the components of the matrix of rotational constraints $\boldsymbol{C}_\theta$ and the components the vector $\boldsymbol{g}_\theta$ are given as

$$\boldsymbol{C}_\theta^{ab}=-\sum_{A=1}^{M}\int_{\partial\mathscr{A}_A^\theta}[h\bar{j}_0^{-1}N_a[p_{b,2}(\boldsymbol{\varphi}_{0,1}\cdot\boldsymbol{\tau}_0)-p_{b,1}(\boldsymbol{\varphi}_{0,2}\cdot\boldsymbol{\tau}_0)]\boldsymbol{t}_0\|\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,t}\|]_A(w_A^Q\circ\boldsymbol{\varphi}_A)\,\mathrm{d}\ell_\xi,$$

$$g_\theta^a=-\sum_{A=1}^{M}\int_{\partial\mathscr{A}_A^\theta}[hN_a\bar{\theta}_\tau\|\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,t}\|]_A(w_A^Q\circ\boldsymbol{\varphi}_A)\,\mathrm{d}\ell_\xi.$$

Note that $\boldsymbol{C}_\theta^{ab}\in\mathbb{R}^{1\times3}$. The tangent boundary vector, defined as $\boldsymbol{\tau}_0=\boldsymbol{\varphi}_{0,t}/\|\boldsymbol{\varphi}_{0,t}\|$, is computed from the expression

$$\boldsymbol{\varphi}_{0,t}=\sum_a\nabla p_a\cdot\boldsymbol{\tau}_\xi\boldsymbol{P}_a,$$

where $\boldsymbol{\tau}_\xi$ is a unit vector tangent to $\partial\mathscr{A}^\theta$.

DOI: 10.1002/nme

ACKNOWLEDGEMENTS

We acknowledge the support of the European Research Council under the European Community's Seventh Framework Programme (FP7/2007-2013)/ERC grant agreement nr. 240487 and the Ministerio de Ciencia e Innovación (DPI2007-61054). M. A. acknowledges the support received through the prize 'ICREA Academia' for excellence in research, funded by the Generalitat de Catalunya.

REFERENCES

1. Alexa M, Behr J, Cohen-Or D, Fleishman S, Levin D, Silva C. Point set surfaces. *VIS '01*: *Proceedings of the Conference on Visualization '01*. IEEE Computer Society: Washington, DC, U.S.A., 2001; 21–28.
2. Pauly M. Point primitives for interactive modeling and processing of 3d geometry. *Ph.D. Thesis*, Federal Institute of Technology (ETH) of Zurich, 2003.
3. Levin D. Mesh-independent surface interpolation. In *Geometric Modeling for Scientific Visualization*, Brunnett H, Mueller (eds). Springer: Berlin, 2003; 37–49.
4. Levoy M, Whitted T. The use of points as displays primitives. *Technical Report TR-85-022*, Computer Science Department, University of North Carolina at Chappel Hill, 1985.
5. Hoppe H, DeRose T, Duchamp T, McDonald J, Stuetzle W. Surface reconstruction from unorganized points. *SIGGRAPH '92*: *Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques*. ACM: New York, NY, U.S.A., 1992; 71–78.
6. Ohtake Y, Belyaev A, Alexa M, Turk G, Seidel H. Multi-level partition of unity implicits. *ACM Transactions on Graphics* (*Proceedings of SIGGRAPH 2003*) 2003; **22**:463–470.
7. Levin D. The approximation power of moving least-squares. *Mathematics of Computation* 1998; **67**(224): 1517–1531.
8. Alexa M, Behr J, Cohen-Or D, Fleishman S, Levin D, Silva C. Computing rendering point set surfaces. *Transactions on Visualization and Computer Graphics* 2003; **9**(1):3–15.
9. Amenta N, Kil Y. Defining point-set surfaces. *ACM Transactions on Graphics* 2004; **23**(3):264–270.
10. Alexa M, Gross M, Pauly M, Pfister H, Stamminger M, Zwicker M. Point-based computer graphics. *SIGGRAPH 2004 Course Notes*, 2004.
11. Krysl P, Belytschko T. Analysis of thin shells by the element-free Galerkin method. *International Journal of Solids and Structures* 1996; **33**(20–22):3057–3078.
12. Noguchi H, Kawashima T, Miyamura T. Element free analyses of shell and spatial structures. *International Journal for Numerical Methods in Engineering* 2000; **47**(6):1215–1240.
13. Chen J, Wang D. A constrained reproducing kernel particle formulation for shear deformable shell in Cartesian coordinates. *International Journal for Numerical Methods in Engineering* 2006; **68**:151–172.
14. Rabczuk T, Areias P, Belytschko T. A meshfree thin shell method for non-linear dynamic fracture. *International Journal for Numerical Methods in Engineering* 2007; **72**:525–548.
15. MacNeal R, Harder R. A proposed standard set of problems to test finite element accuracy. *Finite Element in Analysis and Design* 1985; **1**(1):3–20.
16. Bucalem M, Bathe J. Higher-order Mitc general shell elements. *International Journal for Numerical Methods in Engineering* 1993; **36**:3729–3754.
17. Simo J, Fox D. On a stress resultant geometrically exact shell model. Part I: formulation and optimal parametrization. *Computer Methods in Applied Mechanics and Engineering* 1989; **72**:267–304.
18. Cirak F, Ortiz M, Schröder P. Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *International Journal for Numerical Methods in Engineering* 2000; **47**:2039–2072.
19. Simo J, Fox D, Rifai M. On a stress resultant geometrically exact shell model. Part II: the linear theory; computational aspects. *Computer Methods in Applied Mechanics and Engineering* 1989; **73**:53–92.
20. Yang HTY, Saigal S, Masud A, Kapania RK. A survey of recent shell finite elements. *International Journal for Numerical Methods in Engineering* 2000; **47**(1–3):101–127.
21. Cirak F, Ortiz M. Fully C^1-conforming subdivision elements for finite deformation thin-shell analysis. *International Journal for Numerical Methods in Engineering* 2001; **51**(7):813–833.
22. Engel G, Garikipati K, Hughes T, Larson M, Mazzei L, Taylor R. Continuous/discontinuous finite element approximations of fourth-order elliptic problems in structural and continuum mechanics with applications to thin beams and plates, and strain gradient elasticity. *Computer Methods in Applied Mechanics and Engineering* 2002; **191**:3669–3750.
23. Wells G, Dung N. A C^0 discontinuous Galerkin formulation for Kirchhoff plates. *Computer Methods in Applied Mechanics and Engineering* 2007; **196**(35–36):3370–3380.
24. Noels L, Radovitzky R. A new discontinuous Galerkin method for Kirchhoff–Love shells. *Computer Methods in Applied Mechanics and Engineering* 2008; **197**(33–40):2901–2929.
25. Arroyo M, Ortiz M. Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods. *International Journal for Numerical Methods in Engineering* 2006; **65**:2167–2202.
26. Cyron C, Arroyo M, Ortiz M. Smooth, second order, non-negative meshfree approximants selected by maximum entropy. *International Journal for Numerical Methods in Engineering* 2009; **79**(13):1605–1632.
27. Belytschko T, Stolarski H, Liu W, Carpenter N, Ong J. Stress projection for membrane and shear locking in shell finite-elements. *Computer Methods in Applied Mechanics and Engineering* 1985; **51**:221–258.

DOI: 10.1002/nme

28. Jain AK, Duin R, Mao J. Statistical pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2000; **22**(1):4–37.
29. Zhang Z, Zha H. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM Journal on Scientific Computing* 2005; **26**(1):313–338.
30. Lall S, Krysl P, Marsden J. Structure-preserving model reduction for mechanical systems. *Physica D* 2003; **184**:304–318.
31. Niroomandi S, Alfaro I, Cueto E, Chinesta F. Model order reduction for hyperelastic materials. *International Journal for Numerical Methods in Engineering* 2009; DOI: 10.1002/nme.2733.
32. do Carmo MP. *Differential Geometry of Curves and Surfaces*. Prentice-Hall: Englewood Cliffs, NJ, 1976.
33. Sukumar N. Construction of polygonal interpolants: a maximum entropy approach. *International Journal for Numerical Methods in Engineering* 2004; **61**(12):2159–2181.
34. Hughes T, Cottrell J, Bazilevs Y. Isogeometric analysis: cad, finite elements, nurbs, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering* 2005; **194**:4135–4195.
35. González D, Cueto E, Doblaré M. A higher order method based on local maximum entropy approximation. *International Journal for Numerical Methods in Engineering* 2010; DOI: 10.1002/nme.2855.
36. Rosolen A, Millán D, Arroyo M. On the optimum support size in meshfree methods: a variational adaptivity approach with maximum entropy approximants. *International Journal for Numerical Methods in Engineering* 2009; DOI: 10.1002/nme.2793.
37. Fernández-Méndez S, Huerta A. Imposing essential boundary conditions in mesh-free methods. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**:1257–1275.
38. Gupta MR. An information theory approach to supervised learning. *Ph.D. Thesis*, Stanford, 2003.
39. Stroud AH. *Approximate Calculation of Multiple Integrals*. Prentice-Hall: Englewood Cliffs, NJ, 1971.
40. Ciarlet PG. *Mathematical Elasticity*, *Volume III*: *Theory of Shells*. North-Holland: Amsterdam, 2000.
41. Wang D, Chen J. A hermite reproducing kernel approximation for thin plate analysis with sub-domain stabilized conforming integration. *International Journal for Numerical Methods in Engineering* 2008; **74**:368–390.
42. Green S, Turkiyyah G. Second-order accurate constraint formulation for subdivision finite element simulation of thin shells. *International Journal for Numerical Methods in Engineering* 2004; **61**(3):380–405.
43. Ohtake Y, Belyaev A, Seidel HP. Sparse surface reconstruction with adaptive partition of unity and radial basis functions. *Graphical Models* 2006; **68**(1):15–24.
44. Timoshenko S, Woinowsky-Kreiger S. *Theory of Plates and Shells*. McGraw-Hill: New York, 1959.
45. Lindberg G, Olson M, Cowper G. New developments in the finite element analysis of shells. *Quarterly Bulletin*, Division of Mechanical Engineering, National Aeronautical Establishment, vol. 4, 1969; 1–38.
46. Amenta N, Choi S, Kolluri R. The power crust. *Proceedings of the Sixth ACM Symposium on Solid Modeling and Applications* 2001; 249–260.

Appendix B

Paper # 2: "Nonlinear manifold learning for meshfree finite deformation thin shell analysis"

INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng (2012)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/nme.4403

Nonlinear manifold learning for meshfree finite deformation thin-shell analysis

Daniel Millán, Adrian Rosolen and Marino Arroyo*,†

Laboratori de Càlcul Numèric (LaCàN), Departament de Matemàtica Aplicada III (MA3), Universitat Politècnica de Catalunya (UPC), Campus Nord UPC-C2, E-08034 Barcelona, Spain

SUMMARY

Calculations on general point-set surfaces are attractive because of their flexibility and simplicity in the preprocessing but present important challenges. The absence of a mesh makes it nontrivial to decide if two neighboring points in the three-dimensional embedding are nearby or rather far apart on the manifold. Furthermore, the topology of surfaces is generally not that of an open two-dimensional set, ruling out global parametrizations. We propose a general and simple numerical method analogous to the mathematical theory of manifolds, in which the point-set surface is described by a set of overlapping charts forming a complete atlas. We proceed in four steps: (1) partitioning of the node set into subregions of trivial topology; (2) automatic detection of the geometric structure of the surface patches by nonlinear dimensionality reduction methods; (3) parametrization of the surface using smooth meshfree (here *maximum-entropy*) approximants; and (4) gluing together the patch representations by means of a partition of unity. Each patch may be viewed as a meshfree macro-element. We exemplify the generality, flexibility, and accuracy of the proposed approach by numerically approximating the geometrically nonlinear Kirchhoff–Love theory of thin-shells. We analyze standard benchmark tests as well as point-set surfaces of complex geometry and topology.

Received 23 February 2012; Accepted 3 July 2012

KEY WORDS: shells; meshfree methods; partition of unity; point-set surfaces; *maximum-entropy* approximants; nonlinear dimensionality reduction

1. INTRODUCTION

The Kirchhoff–Love theory of thin-shells is very attractive as compared with shear deformable shells because only the middle surface (three degrees of freedom per node) needs to be described. It has been shown to be a very good model for slender surface-like bodies. However, this theory involves both the first and second fundamental forms of the surface, and consequently, the approximation of the deformation needs to have second-order square integrable derivatives. For general unstructured meshes, it is difficult to define C^1 finite element approximations, which has prompted a myriad of techniques that avoid this requirement ([1–3] and references there in). Yet, C^0 approaches do not perform well in the thin-shell limit.

Early meshfree approaches are among the first numerical methods with smooth approximants for Kirchhoff–Love shells beyond Hermite approximations [4]. Following ideas from computer graphics, smooth subdivision surfaces finite elements have been proposed for thin-shells [5, 6]. Subdivision finite elements provide a unified framework for geometric modeling and thin-shell analysis. Along this line of work, isogeometric analysis [7, 8] is a new technology building on computer-aided design smooth approximation methods, such as B-splines and non-uniform rational

*Correspondence to: Marino Arroyo, Laboratori de Càlcul Numèric (LaCàN), Departament de Matemàtica Aplicada III (MA3), Universitat Politècnica de Catalunya (UPC), Campus Nord UPC-C2, E-08034 Barcelona, Spain.

†E-mail: marino.arroyo@upc.edu

B-splines (NURBS). They have been successfully applied to beams, plates, and thin-shells [8–10]. These methods exhibit a high fidelity of the geometry representation, and their difficulties are derived from the rigidity of the NURBS framework with regards to the structure of the grid, which are the topic of current research [11]. On the other hand, discontinuous Galerkin formulations have been proposed recently for plates, beams, and thin-shells [12–16]. These methods avoid the C^1 continuity requirement by designing suitable numerical fluxes conjugate to the deformation jumps. An advantage of this method is the ease in the imposition of the rotation essential boundary conditions. As disadvantages, the formulation and implementation of these methods is cumbersome, and they typically exhibit a poorer accuracy for a given number of degrees of freedom as compared with methods based on smooth approximants.

Despite the advances made in the area of computer graphics to process point-set surfaces, that is, surfaces discretized merely as a set of points in space [17–24], meshfree methods for thin-shell analysis are still restricted to simple surfaces admitting a single parametrization [4, 25–27]. In a recent work, we presented a methodology based on linear statistical learning techniques to process a general smooth surface defined by a set of points alone [28]. The performance of the method was assessed by confronting a classical obstacle course of linear benchmarks proposed in [29]. The method is in principle applicable to embedded manifolds in any dimension. In our previous work, the method results from combining three ingredients:

1. The local geometric structure of the manifold is detected from the node set using weighted PCA (wPCA), which identifies the hyperplane closest to the points in a given neighborhood that we call patch. The number of patches is in general comparable with the number of points.
2. A smooth local parametrization is defined in the two-dimensional wPCA projection of the points of each patch. This can be realized with a variety of methods, from other meshfree methods such as moving least squares (MLS) approximants to mesh-based methods such as subdivision finite elements. In the latter case, no global mesh is required. Here, the local *maximum-entropy* (*max-ent*) approximants [30] are chosen because of their smoothness, robustness, and relative ease of quadrature compared with other meshfree approximants.
3. The local parametrizations are then glued together with a partition of unity (PU) defined in the ambient space, which consequently is also a PU on the manifold.

In spite of being able to deal with shells of complex topology and geometry, this method presents a serious practical drawback: wPCA produces a massive overlapping between the patches. The patches cannot be too large; otherwise, the local hyperplane approximates poorly the manifold, and the local parametrization may become very distorted or even not injective. The massive overlapping, needed to glue the local patches with the PU, results in great redundancy of quadrature points and a very large computational cost.

In the present work, our goal is to greatly expand the range of applicability of the ideas presented in [28], building more efficient and robust local parametrizations of the point-set surface. We exploit nonlinear dimensionality reduction (NLDR) techniques, instead of the linear PCA. By doing so, the point-set surface can be partitioned into a small number of regions, which exhibit a small overlap. The calculations are significantly faster and more accurate. We illustrate with a numerical example the amount of overlapping with both methods. We consider a scattered set of points distributed on a sphere and compute the ratio between the number of quadrature point needed with a method based on wPCA or NLDR and the number of quadrature points if no overlap at all was needed (Figure 1). It can be observed that the redundancy factor for the method based on wPCA can be as large as 15, whereas with the proposed method, this factor approaches the optimal value of 1 as the discretization is refined.

In Section 2, a concise introduction to dimensionality reduction in the context of data-driven methods is presented. Additionally, two additional steps are proposed to obtain high-quality low-dimensional embeddings, which are motivated with a simple example. Section 3 describes the proposed methodology for point-set manifold processing. Section 4 provides a short account of the geometrically exact Kirchhoff–Love shell theory. Numerical experiments to evaluate the performance of the method are presented in Section 5. Some remarks and conclusions are collected in Section 6.

DOI: 10.1002/nme

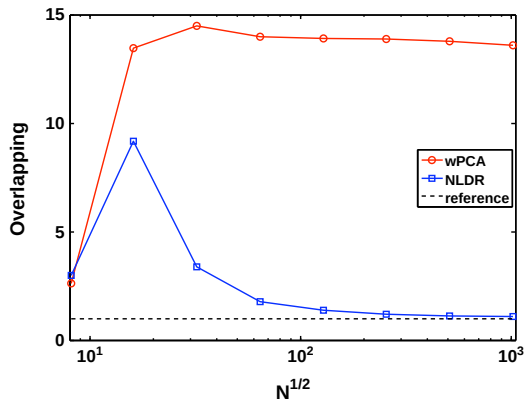


Figure 1. Overlapping ratio comparison between a linear (wPCA) and a nonlinear dimensionality reduction (NLDR) method for a spherical point-set surface. The overlapping measures the amount of redundancy of quadrature points caused by the overlapping of the local parametrizations.

2. DIMENSIONALITY REDUCTION

The goal of statistical learning is extracting meaningful information from empirical data, a general endeavor with many applications in science and engineering. In particular, dimensionality reduction tries to address the curse that dimensionality researchers confront when dealing with high D-dimensional data. It proceeds by finding a lower d-dimensional representation, $d \ll D$, which captures the most relevant features of the data, that is, these methods identify the hidden variables that best explain the behavior of a given system. Generically, given an input data set $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\} \subset \mathbb{R}^D$, the problem is finding a lower dimensional representation $\Xi = \{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_N\} \subset \mathbb{R}^d$ with $d < D$, which in some sense retains the essential information in the original data.

2.1. Linear dimensionality reduction

The most widespread technique of dimensionality reduction was introduced in 1901 by Pearson [31]. In this seminal work, Pearson developed a method to obtain the best affine fit that minimizes the distance from the input data (Figure 2). Throughout the years, this method has been rediscovered and extended in many areas, where it goes by different names: principal components or Hotelling transform [32], Karhunen–Loève transform [33, 34], empirical orthogonal functions [35], and proper orthogonal decomposition [36]. Here, we will refer to it as PCA. See [37] for a modern reference to this subject.

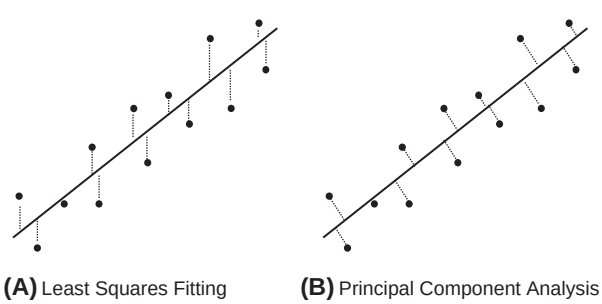


Figure 2. (A) Least squares and (B) PCA fits to a set of scattered points in two dimensions. The least-squares fit depends on the choice of axis and requires solving a linear system of equations, whereas the PCA fit is geometrically objective and requires finding eigenvalues and eigenvectors.

DOI: 10.1002/nme

Principal component analysis is a standard tool in computer graphics [19], data analysis [38], manifold learning [39], or model reduction techniques in computational mechanics [40, 41]. PCA identifies the low d-dimensional subspace that best explains the variance of a higher D-dimensional data set. The original data are transformed to a new orthogonal coordinate system such that the projection of the data on the subspace defined by the first d coordinate directions, $d \leqslant D$, maximizes the variance as compared with any other projection onto a d-dimensional subspace. Given a data set in high dimension $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\} \subset \mathbb{R}^D$, we define the matrix $\boldsymbol{X} \in \mathbb{R}^{D\times N}$, where the centered coordinates of the points, $\boldsymbol{x}_a - \bar{\boldsymbol{x}}$ for $a = 1, \ldots, N$, are placed as column vectors. The average position of all the points $\bar{\boldsymbol{x}}$ is

$$\bar{\boldsymbol{x}} = \frac{1}{N}\sum_{a=1}^{N} \boldsymbol{x}_a.$$

The covariance matrix is then

$$\boldsymbol{C} = \frac{1}{N}\boldsymbol{X}\boldsymbol{X}^{\mathrm{T}} \quad \in \mathbb{R}^{D\times D}.$$

This positive (semi-)definite symmetric matrix has real eigenvalues and diagonalizes in an orthonormal basis of eigenvectors. We define $\boldsymbol{V} \in \mathbb{R}^{D\times d}$ as the eigenvector matrix formed by the d eigenvectors corresponding to the largest d eigenvalues. These vectors span the affine space of dimension d passing through $\bar{\boldsymbol{x}}$, which best describes the data. The matrix $\boldsymbol{V}$ defines an orthogonal projection Π_{PCA} relative to $\bar{\boldsymbol{x}}$ onto the reduced space of dimension d, that is,

$$\begin{aligned} \Pi_{\mathrm{PCA}} : \mathbb{R}^D &\longrightarrow \mathbb{R}^d \\ \boldsymbol{x} &\longmapsto \boldsymbol{V}^{\mathrm{T}}(\boldsymbol{x} - \bar{\boldsymbol{x}}). \end{aligned}$$

The low-dimensional representation of the data is then given by $\boldsymbol{\xi}_a = \Pi_{\mathrm{PCA}}(\boldsymbol{x}_a)$ for $a = 1, \ldots, N$.

Another classical method of linear dimensionality reduction is multidimensional scaling (MDS). MDS is more suitable when, instead of data coordinates, we have similarity scores between each pair of *stimuli*. MDS is a standard tool in psychophysics and sensory analysis. Here, we describe the classical metric MDS [42]. Extensions and details about non-metric MDS can be found for instance in [43] and references therein. The data are given in terms of a full $N \times N$ matrix $\boldsymbol{D}$ containing the squares of the similarities between each pair of points. MDS projects the data set into the affine space that best preserves the similarity scores between the data points. If the similarity score is simply the distance derived from the Euclidean metric, that is, $D_{ab} = |\boldsymbol{x}_a - \boldsymbol{x}_b|^2$, then the method tries to find a nearly isometric low-dimensional embedding of the data, and it can be shown to coincide with PCA. In this case, the Gram matrix formed by the scalar products between pairs of centered data is simply

$$\boldsymbol{S} = \boldsymbol{X}^{\mathrm{T}}\boldsymbol{X} \quad \in \mathbb{R}^{N\times N}.$$

In general, the Gram matrix is obtained from the matrix of squared distances (similarities) $\boldsymbol{D}$ through a double-centering step, $\boldsymbol{S} = -1/2\boldsymbol{H}\boldsymbol{D}\boldsymbol{H}$, where $H_{ij} = \delta_{ij} - 1/N$. The most expensive step in the method is the singular value decomposition of the Gram matrix $\boldsymbol{S} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^{\mathrm{T}}$, where the eigenvalues are sorted in decreasing order. The data points expressed in the latent variables $\Xi = \{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_N\} \subset \mathbb{R}^d$ are computed in matrix form as

$$\boldsymbol{\Xi} = \boldsymbol{Id}_{dN}\boldsymbol{\Lambda}^{1/2}\boldsymbol{U}^{\mathrm{T}} \quad \in \mathbb{R}^{d\times N},$$

where $\boldsymbol{Id}_{dN}$ is a $d \times N$ matrix formed by the first d rows of the $N \times N$ identity matrix. The low-dimensional embedding of out-of-sample points $\boldsymbol{x} \in \mathbb{R}^D$ can be obtained from the MDS projection Π_{MDS} (see [44] for details).

The PCA and MDS share important features: both define a linear implicit mapping from the high-dimensional space into the low-dimensional embedding, and the core operation for both methods is the computation of the eigenvalues and eigenvectors of a full matrix. If the data are not too high dimensional but the number of points N is very large, PCA is preferable in terms of

DOI: 10.1002/nme

memory and computational cost because the covariance matrix is $D \times D$. By contrast, MDS is better suited when dimensionality is very high but the number of points is moderate, as the Gram matrix is $N \times N$.

2.2. Nonlinear dimensionality reduction

In many cases, the high-dimensional data do not conform to an affine subspace but rather to a nonlinear manifold embedded in $\mathbb{R}^D$. Linear methods are not able to capture this manifold and instead detect its affine hull, which can be of significantly higher dimension. This results in a less compact representation of the data, and what is more, the linear representation glosses over the intrinsic and meaningful structure of the data. NLDR methods have emerged over the last decades for the statistical learning of complex data sets with hidden nonlinear structures in areas such as multivariate data analysis [45], pattern recognition [38], and image processing [46, 47] to mention a few. For an extended NLDR literature and insightful remarks, the reader is referred to [44, 48]. Some years ago, two landmark methods have invigorated this field: Isomap [49] and locally linear embedding (LLE) [50, 51]. Each of these papers has collected some 4000 citations over the last decade. They have expanded the range of the applicability of NLDR and have motivated new developments in the field [39, 52–57]. Successful applications of these techniques include climate data analysis [58], the study of the conformation dynamics of molecules [59, 60], and galaxy spectra classification [61].

2.2.1. Isomap. Isomap tries to embed isometrically the high-dimensional data in $\mathbb{R}^d$, where the distance between data points is an approximate geodesic distance on the manifold. It is a mere application of MDS, and the core of the method is in the calculation of the matrix $\boldsymbol{D}$. This method builds a graph whose vertices are the data points and whose links are the union of the k-nearest-neighbor connections for each vertex. The graph is weighted by the Euclidean distance in $\mathbb{R}^D$ between nearby neighbors. The underlying assumption is that, if the manifold is sufficiently well sampled, the k-neighborhood of a point will look Euclidean. Then, the geodesic distance on the manifold is approximated as the shortest path distance on this graph and is used to fill in the matrix of squared distances. Isomap has been shown to be robust (more than LLE) for data polluted with noise or for non-uniformly distributed data points. However, as a corollary of Gauss's *Theorema Egregium* [62], we know that it is not possible to isometrically embed in two dimensions a surface with non-zero Gaussian curvature. This fact leads to a frustration in the algorithm, which can become unstable. As discussed previously for MDS, Isomap has a very high computational cost and memory requirements when the number of points is high but can deal easily with very high-dimensional data.

2.2.2. LLE. Locally linear embedding assumes that the manifold can be locally approximated by a linear small patch, and hence, each point $\boldsymbol{x}_a$, $a = 1, \ldots, N$ can be reconstructed from its nearest neighbors as

$$\boldsymbol{x}_a = \sum_{b \in \mathcal{N}_{\boldsymbol{x}_a}} W_{ab}\, \boldsymbol{x}_b,$$

where $\mathcal{N}_{\boldsymbol{x}_a}$ is the list of indices of the closest k-neighbors to $\boldsymbol{x}_a$ and W_{ab} are the elements of an $N \times N$ sparse matrix containing unknown weights. The sparsity comes from the fact that each point is reconstructed in terms of its nearest neighbors alone. LLE computes these weights by minimizing the cost function

$$\mathcal{E}(\boldsymbol{W}) = \sum_{a=1}^{N} \left| \boldsymbol{x}_a - \sum_{b \in \mathcal{N}_{\boldsymbol{x}_a}} W_{ab}\, \boldsymbol{x}_b \right|^2.$$

In this optimization program, each row of the weight matrix $\boldsymbol{W}$ is constrained to sum up to 1 to enforce the invariance with respect to translations. These linear weights contain the intrinsic information about the local geometrical structure of the manifold. They are invariant with

DOI: 10.1002/nme

respect to translations, scaling, and rotations. Then, with the weights fixed to the solution of the aforementioned program, the low-dimensional embedding is obtained from minimizing

$$\mathcal{F}(\boldsymbol{\Xi}) = \sum_{a=1}^{N} \left| \boldsymbol{\xi}_a - \sum_{b \in \mathcal{N}_{\boldsymbol{x}_a}} W_{ab}\, \boldsymbol{\xi}_b \right|^2 ,$$

subject to $(1/N)\boldsymbol{\Xi}\,\boldsymbol{\Xi}^{\mathrm{T}} = \boldsymbol{I}$. This constraint removes the affine invariance of the solution, by requiring the low-dimensional points to have unit covariance. Thus, this method computes a low-dimensional embedding that respects the local geometric structure contained in the sparse matrix $\boldsymbol{W}$. This minimization problem has a global minimum, which can be solved by finding the smallest eigenvalues and eigenvectors of the $N \times N$ sparse matrix $\boldsymbol{M} = (\boldsymbol{I} - \boldsymbol{W})^{\mathrm{T}}(\boldsymbol{I} - \boldsymbol{W})$. The reader is referred to [50, 51] for full details and to [63] for a concise mathematical description.

Locally linear embedding does not try to be isometric and, in fact, ignores metric information altogether by producing a low-dimensional embedding of unit covariance. It is more sensitive to the number of neighbors than Isomap. From an efficiency viewpoint, in contrast with Isomap, the $N \times N$ matrix whose eigenvalues and eigenvectors need to be computed is sparse for the LLE method. Therefore, this method is applicable to problems with large numbers of sampling points. LLE presents instabilities due to the ill conditioning of the optimization problem that selects the weights, leading often to spurious distortions. The stability of the LLE method has been enhanced in subsequent modifications, such as Hessian LLE [54] or the modified LLE (MLLE) method [57]. These methods produce better-quality embeddings of low dimension in general, and the MLLE method has a small computational overhead as compared with LLE.

2.2.3. Intrinsic dimensionality. In the algorithms outlined earlier, the manifold dimensionality is assumed to be given, and for thin-shells, $d = 2$. Yet, in other applications, the manifold dimensionality may not be known, and we cannot resort to visual inspection in high dimensions. The estimation of the intrinsic dimensionality underlying a high-dimensional data set is an important question in data exploration. There are many methods to estimate d; here, we give a brief account of the correlation dimension method, local PCA, and tracking the reconstruction error [44, Chapter 3].

The correlation dimension method, originally proposed in [64], considers a closed ball of radius ϵ at the center of each data point and counts the number of points inside this ball. The dimension is estimated by noting that the average number of counts should grow linearly with ϵ for 1D objects, quadratically for two-dimensional entities, and so on. Details about practical implementations and insightful remarks can be found in [44]. Local PCA analyzes small local subsets of the data. The intrinsic dimension d for each of these patches is selected such that it preserves a given fraction of the variance of the original data, for example,

$$0.95 \leqslant \frac{\sum_{i=1}^{d} \lambda_i}{\sum_{i=1}^{D} \lambda_i},$$

where λ_i are the eigenvalues of the covariance matrix arranged in decreasing order. In contrast with the global correlation dimension, this method provides a local estimation of the intrinsic dimension. The reconstruction error is a measure of the dissimilarity between the original high-dimensional data points $\boldsymbol{x}_a \in \mathbb{R}^D$ and their reconstruction from the low-dimensional points, generally computed as a weighted average of the neighbors of $\boldsymbol{\xi}_a \in \mathbb{R}^d$. Whereas for PCA, the notion of reconstruction is straightforward, this is not the case for NLDR methods.

The preceding techniques are easy to implement and robust for data not polluted with noise. For noisy measurements, the notion of intrinsic dimension becomes strongly scale dependent, and these methods cannot be used as a black box. In such situations, physical understanding of the system should be combined with dimension estimation methods.

DOI: 10.1002/nme

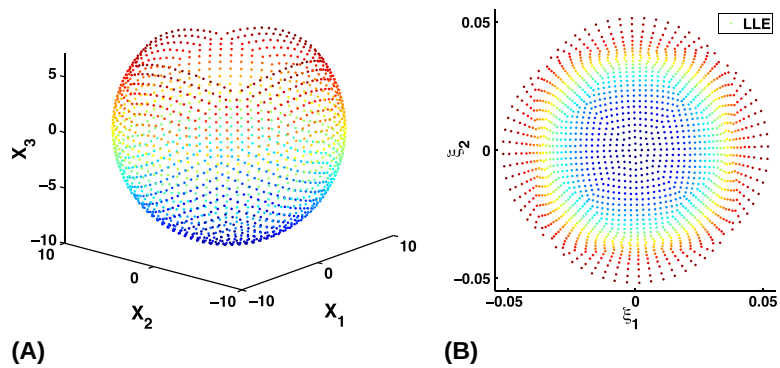


Figure 3. (A) Set of scattered points quasi-uniformly distributed on a truncated sphere. (B) Two-dimensional embedding obtained by locally linear embedding (LLE), exhibiting unavoidable high distortions. The color coding is a visual guide.

2.3. Illustrative examples, point-set partitioning, and affine isometric correction

We show here a few toy examples illustrating the challenges that need to be addressed to use the low-dimensional embeddings given by NLDR techniques as local parametric patches in processing point-set manifolds. We first consider uniformly sampled points lying on a sphere, describing a large portion of the full sphere (Figure 3A). Isomap is not able to embed this cloud of points in two dimensions, despite them clearly representing a surface with the topology of a two-dimensional open set. The frustration arising from the impossibility of embedding isometrically such a large portion of a sphere is too large, and the method fails. LLE provides satisfactory results, at the expense of generating very large local distortions (Figure 3B), which are unavoidable and a corollary of Gauss's *Theorema Egregium*. Another notable feature of the results of LLE in this example is the lack of a clear metric relationship between the low-dimensional embedding and the original data, as noted earlier. Both the distortions and the lack of metric correspondence are problematic for our purposes. Indeed, a good-quality sampling of a surface may become strongly distorted or scaled in one direction if the surface is elongated. As a result, the H^1 norm of the parametrization from the low-dimensional embedding into three dimensions may become very large and non-uniform, leading to inaccurate numerical calculations.

We address the unavoidable distortions by partitioning the original data set into a small number of patches (Figure 4A). We then embed each partition in two dimensions with NLDR techniques. We use the METIS library to partition the data [65]. Not only is partitioning advisable to obtain good-quality embeddings of low dimension, it also becomes unavoidable for surfaces of general topology, as NLDR methods are applicable only to surfaces of trivial topology. It is always possible to recursively partition a data set until all the partitions have trivial topology. Figure 4B–D shows the low-dimensional embedding generated by Isomap, LLE, and MLLE, respectively. Isomap provides a good-quality embedding of low dimension, whereas LLE produces spurious distortions. MLLE corrects these distortions yet introduces a uniform stretching of the domain in one direction as a result of its affine invariance and the unit covariance constraint.

To address the uniform distortions of LLE-based methods, that is, the lack of metric correspondence between the low-dimensional embedding and the original data, we propose a simple iterative method that we call affine isometric correction (AIC). This method finds the affine transformation $\boldsymbol{T}$ that minimizes a measure of isometry [66], given by the discrete stress function

$$E_{\mathrm{S}}(\boldsymbol{T}) = \frac{1}{2}\sum_{a=1}^{N}\sum_{b\in\mathcal{N}_{\boldsymbol{x}_a}}\left(\frac{|\boldsymbol{T}(\boldsymbol{\xi}_a-\boldsymbol{\xi}_b)|-|\boldsymbol{x}_a-\boldsymbol{x}_b|}{|\boldsymbol{x}_a-\boldsymbol{x}_b|}\right)^2,$$

where $\boldsymbol{\xi}_a$ are the low-dimensional points given by a direct application of a NLDR method and $\boldsymbol{T}$ is a $d\times d$ matrix with a positive determinant. By the polar decomposition theorem, it is sufficient to

DOI: 10.1002/nme

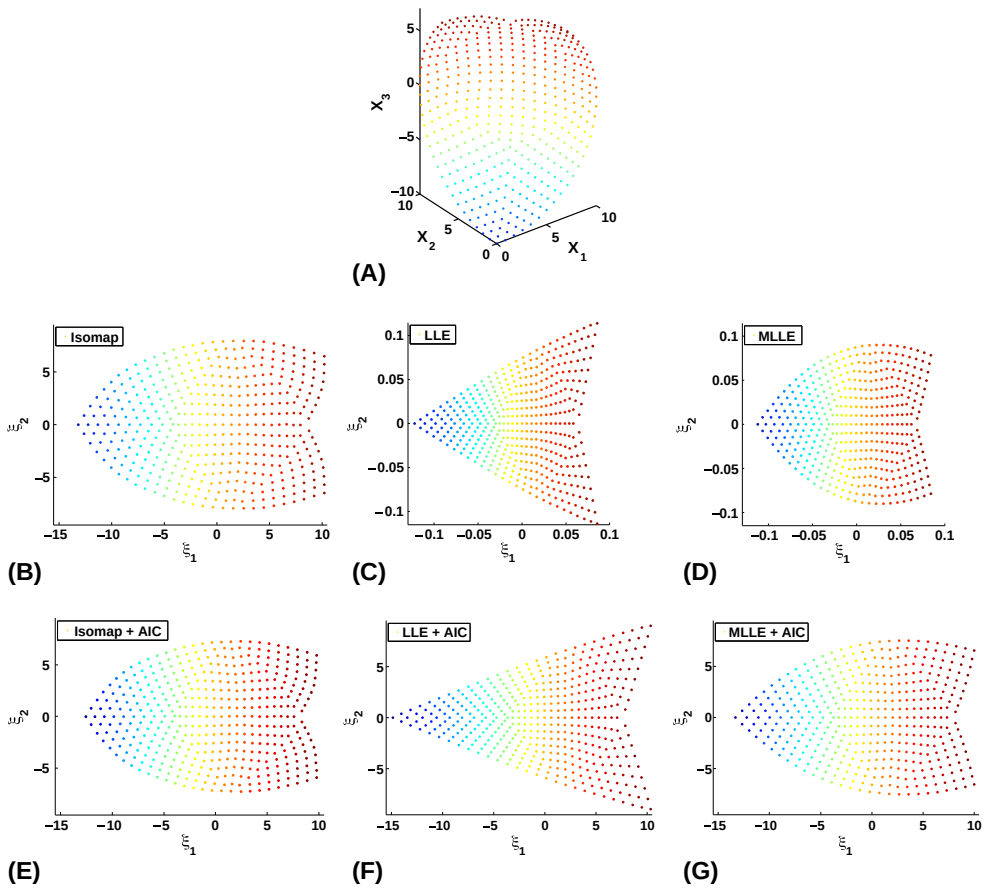


Figure 4. (A) Partition of the data set in Figure 3. Two-dimensional embeddings given by a direct application of (B) Isomap, (C) locally linear embedding (LLE), and (D) modified LLE (MLLE). Two-dimensional embeddings given by (E) Isomap, (F) LLE, and (G) MLLE after the affine isometric correction step.

Table I. Stress function E_S for the quarter of a truncated sphere (Figure 4).

$\boldsymbol{T}$	Isomap	LLE	MLLE
$\boldsymbol{I}$	25.9	30.6×10^4	26.8×10^4
$\arg\min E_S$	11.7	96.5	6.22

Here, we consider a k-rule search for neighbors [67], with $k = 12$.
LLE, locally linear embedding; MLLE, modified locally linear embedding.

minimize the function over symmetric positive-definite matrices. Computationally, we implement Newton's method with line search that converges very rapidly. The low-dimensional embedding resulting from post-processing the raw NLDR results with AIC are shown in Figure 4E–G, and the values of the stress function are given in Table I. It is clear that this simple procedure significantly improves the isometric quality of LLE-based methods *a posteriori*, while it changes only marginally the results of metric methods such as Isomap. In subsequent calculations, we choose the MLLE method combined with the AIC, as it provides good-quality embedding of low dimension metrically related to the original data, and it involves calculations on sparse matrices.

DOI: 10.1002/nme

3. MANIFOLD DESCRIPTION FROM SCATTERED POINTS

3.1. General setting and partition of unity

We consider a smooth d-manifold $\mathcal{M}$ embedded in $\mathbb{R}^D$, $d < D$. Our goal is to numerically represent $\mathcal{M}$ from a set of points and make computations on it. Let $P = \{\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_N\} \subset \mathbb{R}^D$ be a set of control points representing $\mathcal{M}$. We consider another set of geometric markers, $Q = \{\boldsymbol{Q}_1, \boldsymbol{Q}_2, \ldots, \boldsymbol{Q}_M\} \subset \mathbb{R}^D$, typically a subset of P but not necessarily. For simplicity, we will denote the points in P and its associated objects with a lower case subindex, for example, $\boldsymbol{P}_a$, for $a = 1, 2, \ldots, N$, and the geometric markers in Q and its associated objects with an upper case subindex, for example, $\boldsymbol{Q}_A$, for $A = 1, 2, \ldots, M$ $(M \leqslant N)$.

We partition these geometric markers into L groups on the basis of proximity (METIS domain decomposition with a k-nearest neighbor graph). We represent these groups of geometric markers with index sets $\mathcal{I}_\kappa, \kappa = 1, \ldots, L$ with $\cup_{\kappa=1}^{L} \mathcal{I}_\kappa = \{1, 2, \ldots, M\}$ and $\mathcal{I}_\kappa \cap \mathcal{I}_\iota = \emptyset$ when $\kappa \neq \iota$, and use Greek subindices to refer to entities associated with these groups of markers. As it will become clear below, there is a one-to-one correspondence between these groups of geometric markers and the local parametrizations of the surface, which we call patches.

We consider a Shepard PU associated with the geometric markers. Consider a set of non-negative reals $\{\beta_A\}_{A=1,2,\ldots,M}$ associated with each point in Q. We define the Shepard PU with Gaussian weight associated to the set Q as the functions $w_A : \mathbb{R}^D \to \mathbb{R}$ given by

$$w_A(\boldsymbol{x}) = \frac{\exp\left(-\beta_A |\boldsymbol{x} - \boldsymbol{Q}_A|^2\right)}{\sum_{B=1}^{M} \exp\left(-\beta_B |\boldsymbol{x} - \boldsymbol{Q}_B|^2\right)}.$$

For efficiency and given the fast decay of the Gaussian functions, these functions are numerically treated as compactly supported.

We aggregate these PU functions by patches, as it is depicted in Figure 5 for $d = 1$ and $D = 2$, yielding a coarser PU given by the functions

$$\psi_\kappa(\boldsymbol{x}) = \sum_{A \in \mathcal{I}_\kappa} w_A(\boldsymbol{x}).$$

These functions form a PU in $\mathbb{R}^D$ and consequently also in $\mathcal{M}$. We consider the index sets of all control points influencing each patch, $\mathcal{J}_\kappa$, with $\cup_{\kappa=1}^{L} \mathcal{J}_\kappa = \{1, 2, \ldots, N\}$, but now $\mathcal{J}_\kappa \cap \mathcal{J}_\iota \neq \emptyset$ because of the overlap between patch PU functions. Roughly speaking, these sets are $\{a \mid \boldsymbol{P}_a \in \sup \psi_\kappa\}$, slightly enlarged so that the patch parametrization is smooth on the boundary of the support of ψ_κ. Note that, as can be seen in Figure 5, the partition unity functions ψ_κ restricted to the manifold are quite insensitive to the position of the geometrical markers $\boldsymbol{Q}_A$ in the direction normal to the curve. Figure 6 illustrates the overlap regions for a partition of the sphere.

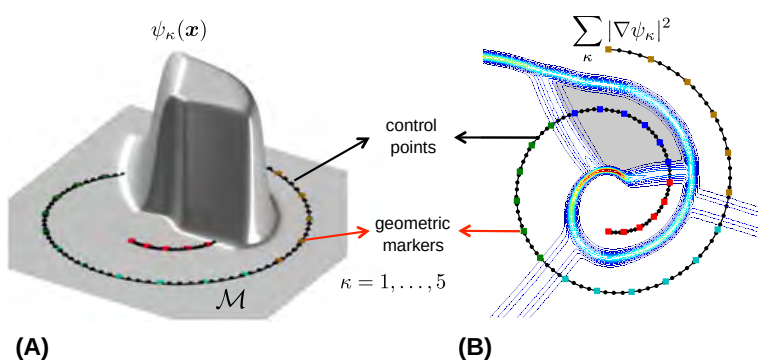


Figure 5. Illustration of the proposed method for a curve ($d = 1$) in the plane $D = 2$. (A) Illustration of a function $\psi_\kappa(\boldsymbol{x})$ of the coarse partition of unity tied to the patches. (B) Visualization of the coarse partition-of-unity overlap regions. The partition of the geometric markers is color coded.

DOI: 10.1002/nme

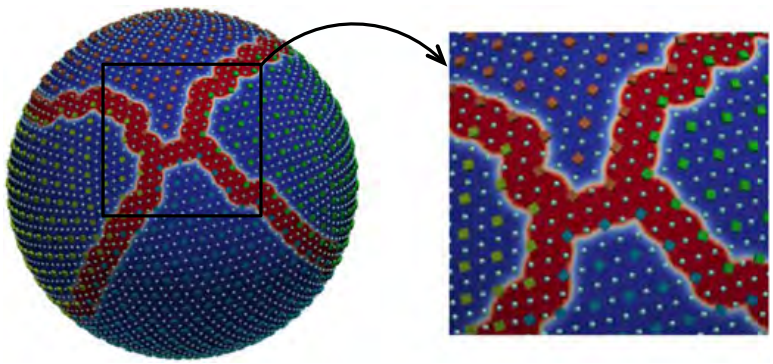


Figure 6. (A) Visualization of a coarse partition of unity overlap regions on a sphere (contour map of $\sum_\kappa |\nabla \psi_\kappa|^2$). (B) Zoom from a selected region on the sphere; the partitioning of the geometric markers ($\diamond$, color coded by the partitions), the width of the overlapping, and the control points $\circ$ can be appreciated.

3.2. Patch manifold learning and surface parametrization

For each patch, we obtain a low-dimensional embedding R_k of the points $P_\kappa = \{\boldsymbol{P}_a\}_{a\in\mathcal{J}_\kappa} \subset \mathbb{R}^D$ with an NLDR technique, resulting in the representation $\Xi_\kappa = \{\boldsymbol{\xi}_a\}_{a\in\mathcal{J}_\kappa} \subset \mathbb{R}^d$, this is

$$\begin{aligned} R_\kappa : \mathbb{R}^D &\longrightarrow \mathbb{R}^d \\ P_\kappa &\longmapsto \Xi_\kappa. \end{aligned}$$

For instance, in meshfree thin-shell analysis, MLLE with AIC produces in general a low-dimensional embedding of good geometric quality and can be computed efficiently. See Figure 7 for an illustration in a complex example. If the automatic partitioning of the data creates patches of complex topology, for example, a tubular partition in an ear of the bunny in the figure or patches of high geometric complexity leading to excessive geometric distortions, such as the blue partition at the tip of one ear, we proceed by recursively re-partitioning such patches.

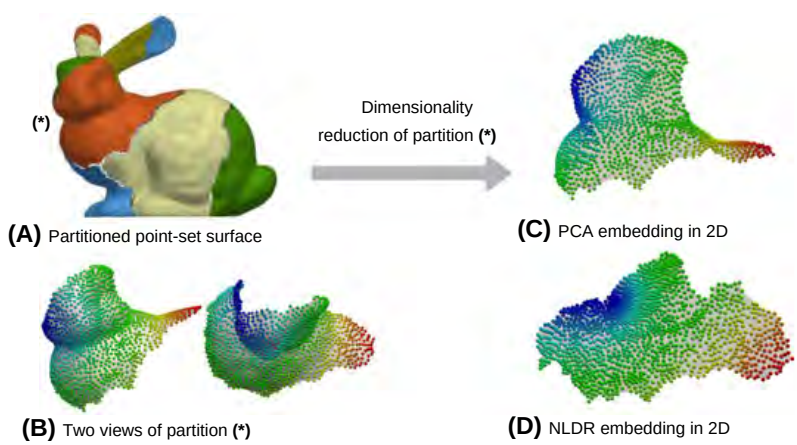


Figure 7. (A) The automatic partitioning of a point-set surface representing the Stanford bunny, performed with METIS, can create patches of complex geometry and topology, for example, a tubular partition in an ear. We recursively partition such patches. (B) Two views of the patch (*). Low-dimensional embedding of the patch (*) by PCA (C) and by a nonlinear method (D). The colors of the points are provided to guide the visual inspection of the embeddings of low dimension. PCA collapses large regions of the patch, whereas the NLDR method successfully 'irons' the curved patch into a moderately distorted low-dimensional embedding.

DOI: 10.1002/nme

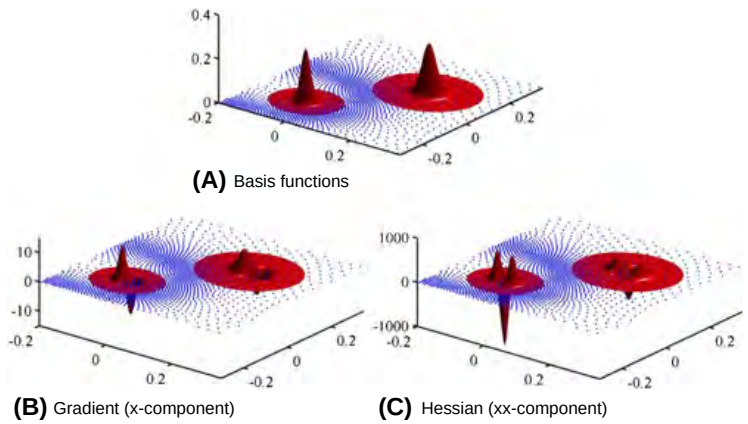


Figure 8. (A) Local *maximum-entropy* basis functions computed with an aspect ratio parameter $\gamma = 0.8$ in an unstructured and non-uniform two-dimensional distribution of points. (B, C) The x-components of the gradient and the Hessian.

The low-dimensional space where the data points of a partition are embedded is a convenient parametric space for the corresponding patch. It is important to note that the embedded points are in general unstructured, and that, although here $d = 2$, the methodology is applicable to higher-dimensional embedded manifolds. This is the topic of current research. The patch parametrizations often need to be smooth, here because of the requirements of the Kirchhoff–Love theory. Thus, a general method to process embedded smooth manifolds demands a smooth approximation scheme for general unstructured nodes in multiple dimensions. There are not many available methods fulfilling these requirements. In two dimensions, subdivision approximations may be used. Here, we consider a general meshfree method to produce such approximants in any spatial dimension d, the local *max-ent* approximants. See [28, 30] for the formulation, properties, and evaluation of the basis functions and their derivatives. The local *max-ent* basis functions in two dimensions are illustrated in Figure 8.

Let $p_a(\boldsymbol{\xi})$ denote the local *max-ent* approximants associated to the point set Ξ_κ on a domain $\mathscr{A}_\kappa \subset \mathbb{R}^d$, a subset of the convex hull of the reduced node set conv Ξ_κ. We locally parametrize the manifold in this patch as

$$\begin{aligned} \boldsymbol{\varphi}_\kappa : \mathscr{A}_\kappa &\longrightarrow \mathbb{R}^D \\ \boldsymbol{\xi} &\longmapsto \sum_{a \in \mathcal{J}_\kappa} p_a(\boldsymbol{\xi})\, \boldsymbol{P}_a. \end{aligned}$$

Note that the images of the local patches, $\boldsymbol{\varphi}_\kappa(\mathscr{A}_\kappa)$, overlap in the vicinity of the partition boundaries and need not exactly coincide in these regions.

3.3. Partition of unity to evaluate integrals on $\mathcal{M}$

A PU is a classical technique to patch together local objects on a manifold [62]. Consider for instance the integral of a scalar function f over a manifold $\mathcal{M}$, $f : \mathcal{M} \to \mathbb{R}$. Then, we have the following identity

$$\int_{\mathcal{M}} f(\boldsymbol{x})\, \mathrm{d}\mathcal{M} = \sum_{\kappa=1}^{L} \int_{\mathcal{M}} \psi_\kappa(\boldsymbol{x})\, f(\boldsymbol{x})\, \mathrm{d}\mathcal{M}.$$

DOI: 10.1002/nme

Combining the PU with the local parametrization of the κth patch, we can approximate numerically integrals over the manifold $\mathcal{M}$ described by a set of scattered points as

$$\int_{\mathcal{M}} f(\boldsymbol{x})\,\mathrm{d}\mathcal{M} \simeq \sum_{\kappa=1}^{L} \int_{\mathscr{A}_\kappa} \psi_\kappa(\boldsymbol{\varphi}_\kappa(\boldsymbol{\xi}))\, f(\boldsymbol{\varphi}_\kappa(\boldsymbol{\xi}))\, J_\kappa(\boldsymbol{\xi})\,\mathrm{d}\boldsymbol{\xi},$$

where $J_\kappa = \sqrt{\det\left[(D\boldsymbol{\varphi}_\kappa)^{\mathrm{T}} D\boldsymbol{\varphi}_\kappa\right]}$ is the Jacobian determinant of the parametrization. In this way, similar to finite element methods, we have split the integral into local contributions, which can be evaluated using local parametrizations. The last integral can be subsequently approximated by a numerical quadrature on the local parametric space. Here, we resort to Gauss quadrature on a support triangulation defined over Ξ_κ.

All variational models describing the mechanics of thin-shells are formulated in terms of integrals over the mid-surface, for example, the elastic energy functional, or the weak form. Consequently, all these theories can be approximated numerically following the preceding ideas. In such cases, the function does not depend explicitly on $\boldsymbol{x}$, but rather on other fields on the manifold, yet the method is still applicable.

3.4. Single representation of a manifold described by multiple patches

We have avoided so far a precise definition of a numerical surface in the overlapping regions. Although a single representation of the manifold given by multiple overlapping patches, which do not coincide exactly at the overlap regions, is not needed to compute integrals and functionals on the manifold, such a representation is useful in a number of situations such as visualization, contact detection, or imposition of displacement at interior points. We describe now how we proceed. As a starting point, we have an out-of-sample point $\boldsymbol{x} \in \mathbb{R}^D$ in the vicinity of the embedded manifold. This point could be a sampling of the actual surface or the image of a point in parametric space by a patch parametrization. Our goal is to define an operator mapping $\boldsymbol{x}$ onto the manifold. We first identify the patches that have an influence on $\boldsymbol{x}$ by building the index set

$$\mathcal{N}_{\boldsymbol{x}} = \{\kappa \in \{1, 2, \ldots, L\} \mid \psi_\kappa(\boldsymbol{x}) > \mathtt{TOL}\}.$$

We can then find the preimage of the closest point projection of $\boldsymbol{x}$ onto the multiple patch representations, which we denote by $\boldsymbol{\xi}_\kappa = \boldsymbol{\varphi}_\kappa^{-1}(\boldsymbol{\pi}_\kappa(\boldsymbol{x}))$ for $\kappa \in \mathcal{N}_{\boldsymbol{x}}$, where $\boldsymbol{\pi}_\kappa(\boldsymbol{x})$ is the closest point projection of $\boldsymbol{x}$ at each patch (Appendix A). We can then define the operator (almost a projection)

$$\boldsymbol{\pi}(\boldsymbol{x}) = \sum_{\kappa \in \mathcal{N}_{\boldsymbol{x}}} \psi_\kappa(\boldsymbol{x})\, \boldsymbol{\varphi}_\kappa(\boldsymbol{\xi}_\kappa(\boldsymbol{x})),$$

which averages the position of the point as represented by the different overlapping parametrizations. Similar to the definition of MLS point-set surfaces [18, 20], we can formally define the numerical surface at overlapping regions as the fixed points of this operator. For MLS point-set surfaces, the properties of the fixed points of a related operator have been mathematically analyzed, and the properties of the resulting manifold established. Our numerical experiments indicate that successive iterations of $\boldsymbol{\pi}(\boldsymbol{x})$ converge extremely fast. In practice, we do not iterate the operator.

4. THIN-SHELL MODEL

In this section, we review the mechanics of thin-shells [5, 6], on the basis of a geometrically exact formulation [3, 68]. We restrict our attention to the Kirchhoff–Love theory of shells, that is, we neglect the shearing and stretching deformation normal to the shell mid-surface. In this theory, the shell director remains normal to the mid-surface during the deformation.

We follow the usual convention for Latin and Greek indices (that is, $i = 1, 2, 3$; $\alpha = 1, 2$), a comma denotes partial differentiation, subscripts refer to covariant components, and superscripts denote contravariant components.

DOI: 10.1002/nme

4.1. Kinematics of the shell

We next describe the kinematics of a thin-shell body $\mathscr{S}$ in three-space (Figure 9). We assume that this body can be described by the pair $(\boldsymbol{\varphi}, \boldsymbol{t})$, where the mapping $\boldsymbol{\varphi}$ defines the shell middle surface, Ω, and $\boldsymbol{t}$ is a field of unit vectors (a field of directors). We assume the thickness h of the shell to be uniform for simplicity, and also we assume that the change in shell thickness after deformation is negligible. Then, the thin-shell body $\mathscr{S}$ is given by

$$\mathscr{S} = \left\{ \boldsymbol{\Phi} \in \mathbb{R}^3 \,|\, \boldsymbol{\Phi} = \boldsymbol{\varphi}(\xi^\alpha) + \xi\, \boldsymbol{t}(\xi^\alpha),\ -\frac{h}{2} \leqslant \xi \leqslant \frac{h}{2},\ (\xi^1, \xi^2) \in \mathscr{A} \right\},$$

where $\mathscr{A} \subset \mathbb{R}^2$ is the parametric space for the middle surface. Hence, we view a configuration $\boldsymbol{\Phi}$ as a mapping from a parametric domain $\mathscr{A} \times [-h/2, h/2]$ into $\mathbb{R}^3$. The parametric domain is described by the coordinates $\{\xi^1, \xi^2, \xi^3\}$ (where we identify $\xi = \xi^3$), whose corresponding dual basis is $\{\boldsymbol{E}^i\}$. The area element of the middle surface can be computed as $d\Omega = \bar{j}\, d\xi^1 d\xi^2$, where $\bar{j} = |\boldsymbol{\varphi}_{,1} \times \boldsymbol{\varphi}_{,2}|$. The tangent map of a given configuration $T\boldsymbol{\Phi}$ can be computed from the convective basis vectors $\boldsymbol{g}_i$ as

$$T\boldsymbol{x} = \frac{\partial \boldsymbol{\Phi}}{\partial \xi^i} \otimes \boldsymbol{E}^i = \boldsymbol{g}_i \otimes \boldsymbol{E}^i,$$

with $\boldsymbol{g}_\alpha = \partial\boldsymbol{\Phi}/\partial\xi^\alpha = \boldsymbol{\varphi}_{,\alpha} + \xi\, \boldsymbol{t}_{,\alpha}$ and $\boldsymbol{g}_3 = \partial\boldsymbol{\Phi}/\partial\xi = \boldsymbol{t}$. The covariant components of the metric tensor in convected coordinates are given by $g_{ij} = \boldsymbol{g}_i \cdot \boldsymbol{g}_j$.

The subscript 0 denotes quantities in the reference configuration, for instance, $\boldsymbol{\varphi}_0$ parametrizes the reference middle surface. A deformation mapping is a mapping from a reference body into $\mathbb{R}^3$, $\boldsymbol{\Phi} \circ \boldsymbol{\Phi}_0^{-1}$. Consequently, the deformation gradient is $\boldsymbol{F} = T\boldsymbol{\Phi}\ (T\boldsymbol{\Phi}_0)^{-1}$, and the Jacobian is $J = \det(\boldsymbol{F}) = j/j_0$, where $j = \det(T\boldsymbol{\Phi}) = \boldsymbol{g}_3 \cdot (\boldsymbol{g}_1 \times \boldsymbol{g}_2)$.

The shell director in the reference configuration $\boldsymbol{t}_0$ coincides with the normal to the undeformed middle surface of the shell and hence

$$\boldsymbol{t}_0 = \frac{\boldsymbol{\varphi}_{0,1} \times \boldsymbol{\varphi}_{0,2}}{\bar{j}_0}, \quad \boldsymbol{\varphi}_{0,\alpha} \cdot \boldsymbol{t}_0 = 0, \quad |\boldsymbol{t}_0| = 1, \quad \boldsymbol{t}_0 \cdot \boldsymbol{t}_{0,\alpha} = 0.$$

In general, the director in the deformed configuration of the shell, $\boldsymbol{t}$, is allowed to be an arbitrary vector field over $\Omega = \boldsymbol{\Phi}(\mathscr{A} \times \{0\})$.

The local shell deformations can be characterized by the Green–Lagrange strain tensor. Because the convected components of the metric tensor coincide with the components of $(T\boldsymbol{\Phi})^{\mathrm{T}} T\boldsymbol{\Phi}$ in the basis associated with $\{\xi^i\}$, the Green–Lagrange strain tensor can be expressed as the difference between the metric tensors on the deformed and undeformed configurations of the shell, that is,

$$E_{ij} = \frac{1}{2}(g_{ij} - g_{0ij}) = \frac{1}{2}(\boldsymbol{\Phi}_{,i} \cdot \boldsymbol{\Phi}_{,j} - \boldsymbol{\Phi}_{0,i} \cdot \boldsymbol{\Phi}_{0,j}).$$

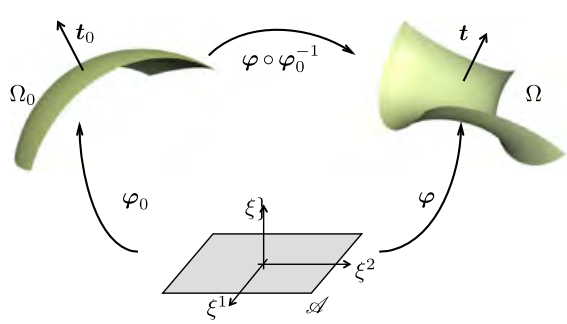


Figure 9. Reference, deformed and parametric configurations of the middle shell surface.

DOI: 10.1002/nme

Plugging the basic kinematic ansatz $\boldsymbol{\Phi} = \boldsymbol{\varphi}(\xi^\alpha) + \xi\ \boldsymbol{t}(\xi^\alpha)$ into the preceding expression and grouping terms, we obtain

$$E_{ij} = \varepsilon_{ij} + \xi\ \rho_{ij} + (\xi)^2\ \vartheta_{ij},$$

which admits the following interpretation in terms of the symmetric tensors ε_{ij}, ρ_{ij}, and ϑ_{ij}:

- The membrane strain tensor $\varepsilon_{\alpha\beta} = (1/2)(\boldsymbol{\varphi}_{,\alpha}\cdot\boldsymbol{\varphi}_{,\beta} - \boldsymbol{\varphi}_{0,\alpha}\cdot\boldsymbol{\varphi}_{,\beta})$, which lives on the middle surface, measures the in-plane deformation of the surface; the components $\varepsilon_{\alpha 3} = (1/2)\boldsymbol{\varphi}_{,\alpha}\cdot\boldsymbol{t}$ measure the shearing of the director $\boldsymbol{t}_0$; and the component $\varepsilon_{33} = (1/2)(\boldsymbol{t}\cdot\boldsymbol{t} - 1)$ measures the stretching of the director $\boldsymbol{t}_0$.
- The bending or change in curvature of the shell is measured by the tensor $\rho_{\alpha\beta} = \boldsymbol{\varphi}_{,\alpha}\cdot\boldsymbol{t}_{,\beta} - \boldsymbol{\varphi}_{0,\alpha}\cdot\boldsymbol{t}_{0,\beta}$, and $\rho_{\alpha 3} = (1/2)\boldsymbol{t}_{,\alpha}\cdot\boldsymbol{t}$ measures the shearing originating from the director elongation; the in-plane tensor $\vartheta_{\alpha\beta} = (1/2)(\boldsymbol{t}_{,\alpha}\cdot\boldsymbol{t}_{,\beta} - \boldsymbol{t}_{0,\alpha}\cdot\boldsymbol{t}_{0,\beta})$ is exclusively related to changes of the middle-surface directors. The rest of the components vanish, $\rho_{33} = \vartheta_{3i} = \vartheta_{i3} = 0$.

4.2. *Kirchhoff–Love hypothesis*

In the remainder of this section, we restrict our attention to the Kirchhoff–Love theory of thin-shells, that is, we constrain the deformed director $\boldsymbol{t}$ to coincide with the unit normal of the deformed middle surface of the shell, that is,

$$\boldsymbol{t} = \frac{\boldsymbol{\varphi}_{,1}\times\boldsymbol{\varphi}_{,2}}{\bar{j}}, \quad \boldsymbol{\varphi}_{,\alpha}\cdot\boldsymbol{t} = 0, \quad |\boldsymbol{t}| = 1, \quad \boldsymbol{t}\cdot\boldsymbol{t}_{,\alpha} = 0.$$

This assumption is well suited when the ratio between the shell thickness and its characteristic size is $\ll 1$. With the Kirchhoff–Love hypothesis, the only remaining non-zero components of the Green–Lagrange strain tensor are

$$\begin{aligned} E_{\alpha\beta} &= \frac{1}{2}(a_{\alpha\beta} - a_{0\alpha\beta}) + \xi(\kappa_{\alpha\beta} - \kappa_{0\alpha\beta}) + \frac{(\xi)^2}{2}(\boldsymbol{t}_{,\alpha}\cdot\boldsymbol{t}_{,\beta} - \boldsymbol{t}_{0,\alpha}\cdot\boldsymbol{t}_{0,\beta}) \\ &= \varepsilon_{\alpha\beta} + \xi\ \rho_{\alpha\beta} + (\xi)^2\ \vartheta_{\alpha\beta}, \end{aligned} \tag{1}$$

where we have introduced the first and second fundamental forms expressed in convected components

$$\begin{aligned} a_{\alpha\beta} &= \boldsymbol{\varphi}_{,\alpha}\cdot\boldsymbol{\varphi}_{,\beta}, \\ \kappa_{\alpha\beta} &= \boldsymbol{\varphi}_{,\alpha}\cdot\boldsymbol{t}_{,\beta} = -\boldsymbol{\varphi}_{,\alpha\beta}\cdot\boldsymbol{t}. \end{aligned}$$

Thus, the Kirchhoff–Love kinematic assumption leads to a formulation of the shell exclusively in terms of the middle surface.

4.3. *Equilibrium configurations*

The potential energy of an elastic shell body with bulk internal energy density W can be expressed as

$$\Pi[\boldsymbol{\Phi}] = \int_{\mathscr{S}_0} W(E_{ij})\ \mathrm{d}V_0 + \Pi_{\text{ext}}[\boldsymbol{\Phi}],$$

where Π_{ext} is the potential energy of the external loads. For concreteness, we consider an isotropic Kirchhoff–St. Venant elastic material [69]

$$W = \frac{1}{2}C^{ijkl}E_{ij}E_{kl},$$

where C^{ijkl} are the contravariant components of the elasticity tensor.

For thin-shell bodies, the Green–Lagrange tensor components are commonly retained up to first order in h (Equation (1)), and the effect of curvature on the Jacobian away from the middle surface is neglected, that is, $j_0/\bar{j}_0 = 1$ [3, 68]. Assuming that the elasticity tensor does not vary through

DOI: 10.1002/nme

the thickness, the internal energy density can be integrated through the thickness, resulting in an internal energy density per unit area

$$\mathcal{W}(E_{\alpha\beta}) = \frac{1}{2}\int_{-h/2}^{h/2} C^{\alpha\beta\gamma\delta} E_{\alpha\beta} E_{\gamma\delta} \frac{j_0}{\bar{j}_0}\, \mathrm{d}\xi \simeq \frac{1}{2} C^{\alpha\beta\gamma\delta}\left(h\varepsilon_{\alpha\beta}\varepsilon_{\gamma\delta} + \frac{h^3}{12}\rho_{\alpha\beta}\rho_{\gamma\delta}\right),$$

with

$$C^{\alpha\beta\gamma\delta} = \frac{E}{(1-\nu^2)}\left[\nu a_0^{\alpha\beta} a_0^{\gamma\delta} + \frac{1}{2}(1-\nu)\left(a_0^{\alpha\gamma} a_0^{\beta\delta} + a_0^{\alpha\delta} a_0^{\beta\gamma}\right)\right],$$

where $a_0^{\alpha\gamma}(a_0)_{\gamma\beta} = \delta^\alpha_\beta$, E is Young's modulus, and ν is Poisson's ratio. Thus, the internal potential energy is a functional of the middle-surface configuration, which can be written as an integral over the reference middle surface

$$\Pi_{\text{int}}[\boldsymbol{\varphi}] = \int_{\Omega_0} \mathcal{W}(E_{\alpha\beta})\, \mathrm{d}\Omega_0,$$

and the external potential becomes

$$\Pi_{\text{ext}}[\boldsymbol{\varphi}] = -\int_{\Omega_0} \boldsymbol{q}\cdot\boldsymbol{\varphi}\, \mathrm{d}\Omega_0 - \int_{\partial\Omega_0} \boldsymbol{h}\cdot\boldsymbol{\varphi}\, \mathrm{d}\ell_0,$$

where $\boldsymbol{q}$ is the external body load per unit area, $\boldsymbol{h}$ are the forces per unit length applied on the boundary of the middle surface, and $d\ell_0$ is the line element of the boundary of the middle surface. Distributed torques can be also applied at the boundary of the thin-shell.

Following [3], we introduce the elastic constitutive relations between the shell stresses and the strains as

$$n^{\alpha\beta} = \frac{\partial \mathcal{W}}{\partial \varepsilon_{\alpha\beta}} = h\, C^{\alpha\beta\gamma\delta}\varepsilon_{\gamma\delta},$$
$$m^{\alpha\beta} = \frac{\partial \mathcal{W}}{\partial \rho_{\alpha\beta}} = \frac{h^3}{12} C^{\alpha\beta\gamma\delta}\rho_{\gamma\delta},$$

where $n^{\alpha\beta}$ is the membrane stress resultant and $m^{\alpha\beta}$ is the bending stress resultant.

The stable equilibrium configurations of the shell minimize the total potential energy, subject to the boundary conditions, and, consequently, satisfy the principle of virtual work, expressed here in terms of integrals over the parametric space $\mathscr{A}$:

$$0 = \delta\Pi[\boldsymbol{\varphi}, \delta\boldsymbol{\varphi}] = \int_{\mathscr{A}} (\delta\boldsymbol{\varepsilon}\cdot\boldsymbol{n} + \delta\boldsymbol{\rho}\cdot\boldsymbol{m})\, \bar{j}_0\, \mathrm{d}\xi^1\mathrm{d}\xi^2 + \delta\Pi_{\text{ext}}[\delta\boldsymbol{\varphi}].$$

4.4. Ritz–Galerkin discretization

We consider now the discrete equilibrium equations for a shell whose middle surface in the reference configuration is numerically represented with the procedure described before, in terms of a set of nodes $P_0 = \{\boldsymbol{P}_{01}, \ldots, \boldsymbol{P}_{0N}\}$ and a set of L patches. We follow a total Lagrangian approach, with the same parametric space and basis functions for the reference and deformed configurations. Let $\boldsymbol{\varphi}_{0\kappa}$ be the reference configuration mapping for the middle surface of a specific patch κ, defined over the parametric space $\mathscr{A}_\kappa$

$$\boldsymbol{\varphi}_{0\kappa}(\boldsymbol{\xi}) = \sum_{a\in\mathcal{J}_\kappa} p_a(\boldsymbol{\xi})\, \boldsymbol{P}_{0a},$$

DOI: 10.1002/nme

as described in Section 3.2. We represent the deformed configuration in a given patch κ as

$$\boldsymbol{\varphi}_\kappa(\boldsymbol{\xi}) = \sum_{a \in \mathcal{J}_\kappa} p_a(\boldsymbol{\xi})\ \boldsymbol{P}_a.$$

With the strategy presented in Section 3.3 and the preceding definitions, the internal elastic energy of the discretized shell can be split into patch contributions

$$\Pi^h_{\text{int}}(\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_N) = \sum_{\kappa=1}^{L} \int_{\mathscr{A}_\kappa} (\psi_\kappa \circ \boldsymbol{\varphi}_{0\kappa})\ \mathcal{W}(E_{\alpha\beta})\ \bar{j}_0\ \mathrm{d}\xi^1 \mathrm{d}\xi^2,$$

where $E_{\alpha\beta}$ and $\bar{j}_0$ are evaluated with the κth patch approximation of the undeformed and deformed configurations. Note that only $E_{\alpha\beta}$ depends on the unknown control points defining the deformed configuration. The external potential is numerically computed likewise. Equilibrium configurations satisfy that the out-of-balance forces vanish

$$0 = \frac{\partial \Pi^h}{\partial \boldsymbol{P}_a}(P) = \boldsymbol{f}^a_{\text{int}}(P) - \boldsymbol{f}^a_{\text{ext}}(P).$$

Stable equilibrium configurations are obtained by numerically minimizing $\Pi^h(P)$, where the essential displacement and rotation boundary conditions are imposed with Lagrange multipliers in an augmented Lagrangian framework. Within the augmented Lagrangian loop, we first obtain a coarse and robust approximation of the equilibrium point with a limited-memory Broyden–Fletcher–Goldfarb–Shanno method and then switch to Newton's method combined with line search to refine the minimization. Details about the calculation of the out-of-balance forces, the tangent stiffness matrix, the boundary constraints, and the solution method are given in Appendices C, D, and E, respectively.

5. NUMERICAL EXAMPLES

We exercise the proposed method with some standard numerical linear and nonlinear benchmark tests. For the linear analysis of thin-shells, we consider the classical problem of a hemisphere loaded with two pairs of facing concentrated forces [29]. Then, we analyze two popular nonlinear problems [70], an open hemispherical shell subjected to alternating radial forces and the pullout of a cylindrical shell with open ends. Finally, the flexibility of the proposed methodology to deal with shells of complex topology and geometry is illustrated by two additional examples at the end of this section. A collection of videos highlighting the nonlinear mechanics of these geometrically exact shells can be found at [71].

5.1. Numerical aspects

We refer to [28] for a detailed account on the max-ent basis functions and the numerical parameters involved. We only note here, from this reference, that linearly reproducing local *max-ent* approximants with relatively wide support can very accurately approximate thin-shell problems with functionals involving second-order derivatives. The smoothness or aspect ratio of the basis functions is controlled by a nondimensional parameter, γ_{LME}. We choose $\gamma_{\text{LME}} = 0.8$, which provides accurate solutions at a moderate computational cost. Similarly, for the PU Shepard functions, we select $\gamma_{\text{PU}} = 4.0$, which results in moderately narrow overlap regions. In all the examples, we build a Delaunay triangulation in the low-dimensional embedding of each meshfree macro-element and generate a standard Gauss–Legendre cubature rule of 12 points (order 6) per triangle, an overkill integration rule. At the boundary curves, to integrate the boundary constraints, we choose a quadrature scheme of four Gauss–Legendre points per integration cell. As in [28, 72], we resort to ghost nodes at the boundaries of the middle surface to avoid the loss of accuracy caused by the excessive flattening of the *max-ent* approximants at the boundary of the convex hull of the nodes. The number of marker points is one order of refinement lower than the number of control points.

DOI: 10.1002/nme

Smooth convex approximants, such as local *max-ent* methods, B-splines, and NURBS basis functions, are in general not interpolating. Therefore, if the set of control points lies on the manifold, a systematic error is introduced, which for shells generally results in a stiffer behavior. Here, we fit the control points so that the reconstruction error of the original surface is minimized in a least-squares sense (Appendix B). This procedure improves the accuracy of the method, although it is not required for optimal convergence rate, as shown in the next example.

5.2. *Pinched closed hemisphere*

In this example, a hemispherical shell of radius $R = 10$ and thickness $h = 0.04$ is subjected to two pairs of radial loads $F = 2$ acting along diametral directions (Figure 10A). This is a challenging test, which assesses the method's ability to represent inextensional deformations under complex shell bending conditions with curvature in two directions. The convergence of the relative error for the radial displacement is shown in Figure 10B. The displacements are normalized by a deflection of $\delta_r = 0.09241$ obtained by an overkill calculation, which agrees with the lower bound given in [1, 29], that is, $\delta_r = 0.0924$. In this figure, we plot the convergence results reported in [72] for subdivision finite elements based on Loop's scheme (triangular elements) and on Catmull–Clark's scheme (quadrilateral elements), as well as results with the previous version of our method reported in [28]. The excellent convergence properties of the proposed method is clear from the figure. We obtain more accurate results for a given number of degrees of freedom than arguably the most competitive method for thin-shells. Our method is more expensive than subdivision finite elements because of the quadrature and the larger sparsity pattern of the stiffness matrix, which makes a full comparison difficult. We can also see that the results of the present method are very similar to

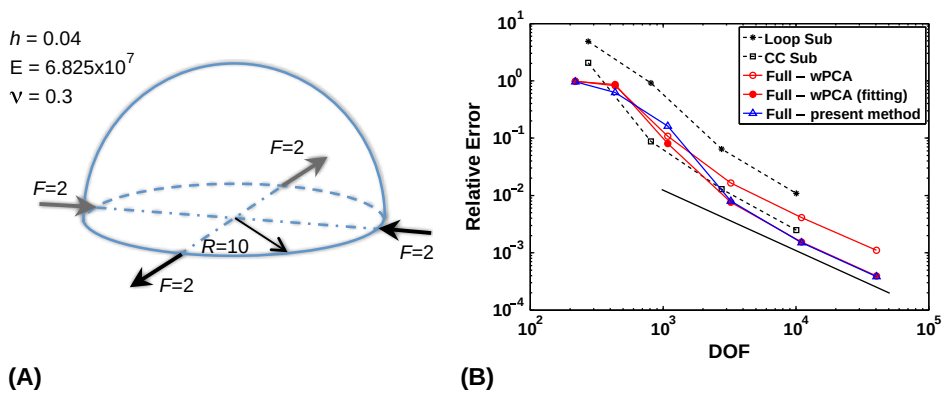


Figure 10. (A) Sketch of the pinched closed hemisphere shell test. (B) Convergence of the normalized radial displacement for two subdivision schemes [72], for the weighted PCA (wPCA) method proposed in [28] and for the present method based on modified locally linear embedding.

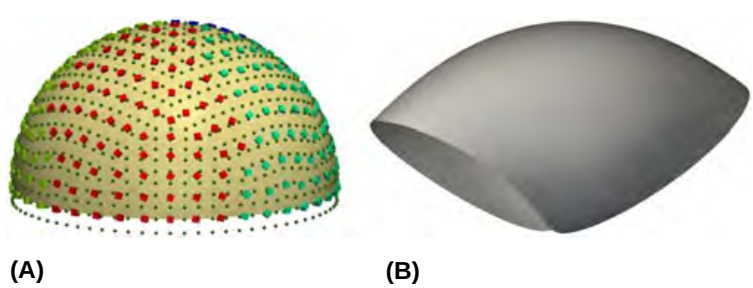


Figure 11. Pinched hemisphere: geometric markers ◇, control points ○, and reference configuration (A) and deformed surface (B). In this linear problem, the deformation has been amplified by a factor of 100.

DOI: 10.1002/nme

those obtained with our previous local linear manifold learning method, which nevertheless is much more expensive, as discussed earlier (Figure 1). Finally, we note the effect of fitting the control points to better represent the geometry. The control and patch points and the deformation are shown in Figure 11.

5.3. *Pinched open hemisphere*

This is a classical nonlinear benchmark analyzing a hemispherical shell with a hole of 18° in its pole. The shell is pinched by two opposite pairs of forces, as shown in Figure 12A. Figure 12B plots the radial displacements under the loads. We compare our results against the results given by ABAQUS's S4R four-node shell element [70]. The agreement is remarkable.

5.4. *Pullout of an open-ended cylindrical shell.*

It is a challenging nonlinear benchmark, which shows the capabilities of the method to deal with problems with comparable membrane and bending energies. The material and geometrical properties for this benchmark are indicated in Figure 13A, whereas the radial displacements of the test points A, B, and C are plotted in Figure 13B. Again, we compare our results against those in [70], and the agreement is excellent. The biggest difference is at the indentation points.

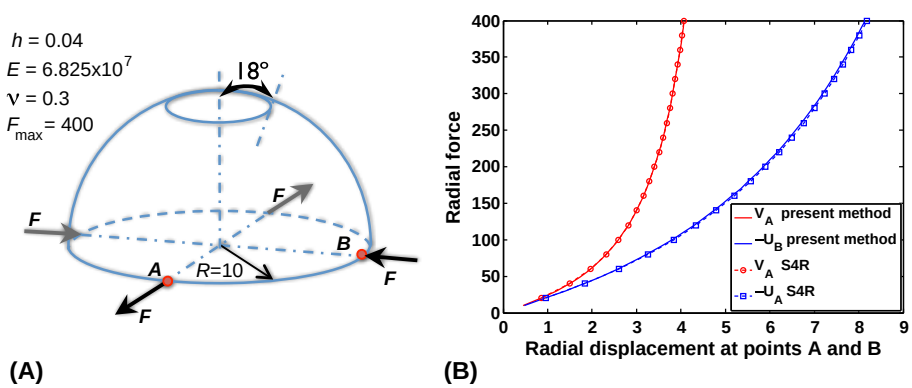


Figure 12. (A) Sketch for the pinched open hemisphere problem. (B) Load–deflection curves for the open hemispherical shell subjected to two pairs of facing concentrated forces.

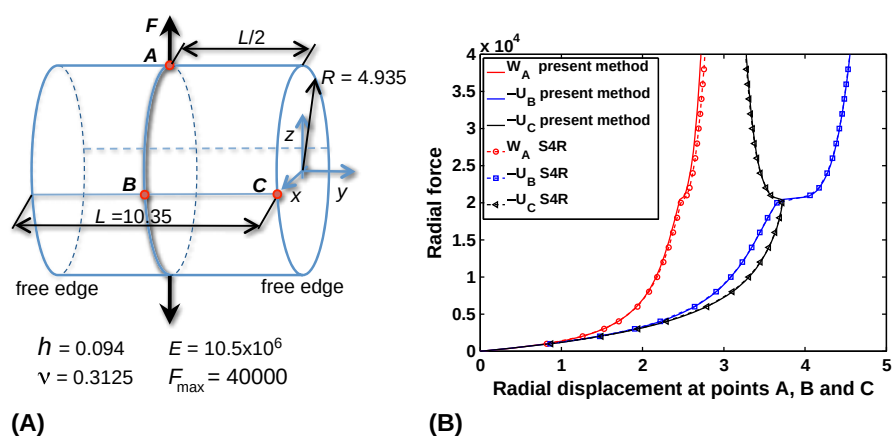


Figure 13. (A) Sketch for the pulled cylinder with free edges. (B) Load–deflection curves of an open-ended cylindrical shell under radial forces.

DOI: 10.1002/nme

We interpret this small disagreement as an overestimation of the S4R elements caused by their non-smooth interpolating character. Figure 14 shows selected snapshots along the deformation, illustrating the buckling event for a force of around $2 \cdot 10^4$.

Figure 14. Selected snapshots of the deformation process during the pullout of an open-ended cylindrical shell (deformation not magnified), showing the buckling event for a force of about $2 \cdot 10^4$.

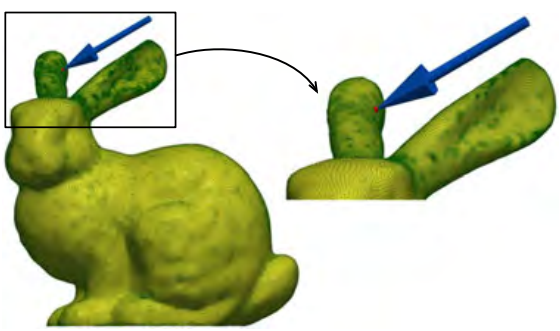


Figure 15. Control points of the bunny and depiction of the imposed displacement, which moves along the blue arrow.

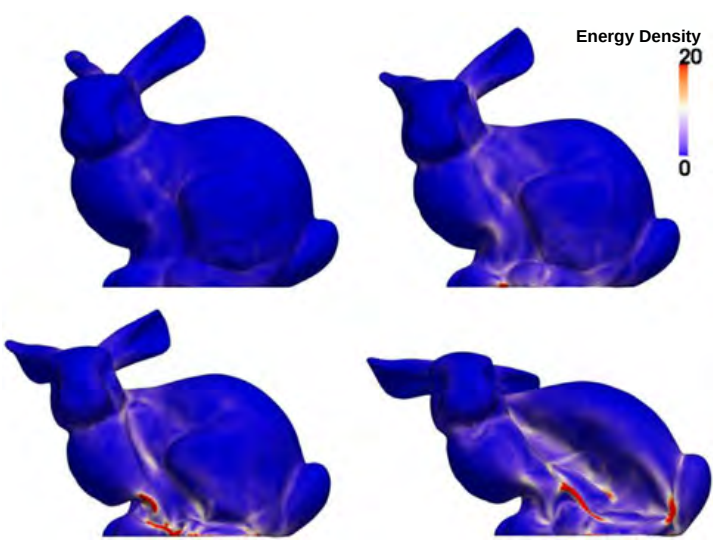


Figure 16. Elastic energy density at selected equilibrium configurations for the bunny thin-shell under a static incremental loading.

DOI: 10.1002/nme

5.5. *Indentation of a bunny*

We illustrate now the ability of the method to deal with shells of complex geometry defined by a set of points alone, without the need for a global surface mesh. The only requirement on the nodes is that they sample sufficiently well the geometric features. This problem is treated with the full geometric nonlinearity. The height of the object is around 1.5, and the thickness of the thin-shell is $h = 0.005$. The material parameters are $E = 10^7$ and $\nu = 0.3$. Figure 15 shows a sketch of 54,867 control points sampling the Stanford bunny [73], which is deformed by an imposed displacement moving incrementally in the direction of the blue arrow.

Figure 16 shows four snapshots along the deformation process, experiencing a number of buckling events (see [71] for illustrative movies). The deformation is not magnified. The simulation proceeds robustly and exhibits very large deformations and localized creases with strain energy density concentrations, typical of the post-buckling response of thin-shells.

5.6. *Connected pipes*

We now illustrate the ability of the proposed method to deal with extremely complex topologies (Figure 17). The boundary curves of the bottom pipes are clamped, and the top boundary curves

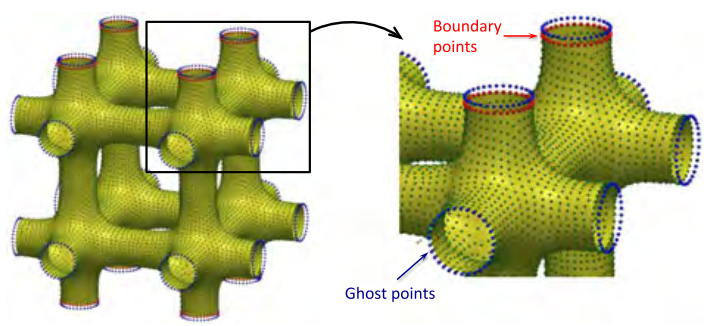


Figure 17. Point-set for the example of the connected pipes.

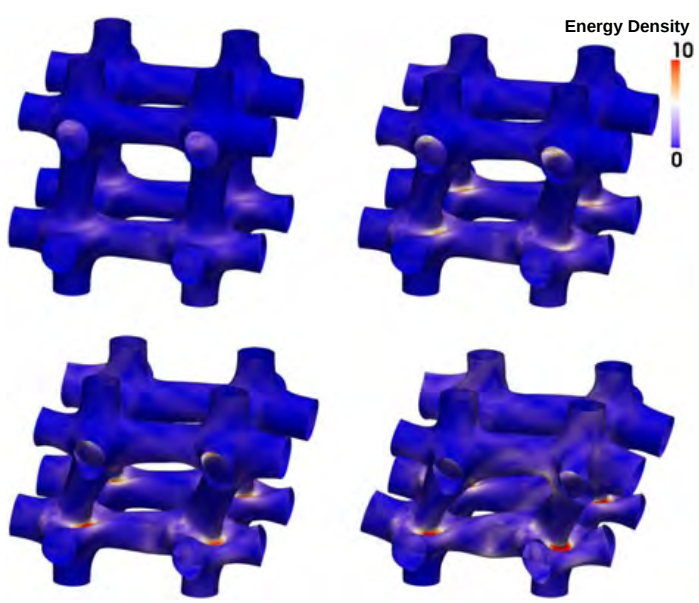


Figure 18. Selected snapshots of the deformation process of the shell with complex topology. Nonmagnified deformation and color map of the strain energy density.

DOI: 10.1002/nme

are incrementally displaced in the $(-1,-1,-1)$ direction. The lateral dimension of the system is 6.2, and the shell thickness is $h = 0.03$. Figure 17 illustrates the discretization with 14,176 control points. Figure 18 shows the energy density on the deformed configuration at four snapshots, without magnification of the displacements. Again, the shell undergoes several geometric instabilities and exhibits localized elastic deformations. We insist on the fact that this thin-shell cannot be studied with previous meshfree approaches because it does not admit a single parametric space.

6. CONCLUSIONS

We have extended the methodology proposed in [28] to build smooth numerical representations of d-dimensional point-set manifolds embedded in $\mathbb{R}^D$, which avoids a global parametrization or a mesh. The proposed method exploits modern NLDR techniques, such as Isomap and LLE, to find embeddings of low dimension of good geometric quality for large patches of surface defined by automatic partitioning of the set of points. A meshfree parametrization of the patches is then defined, with local *max-ent* approximants. The different patches can be glued together with a PU associated to the patches, which allows us to split the evaluation of functionals on the manifold in patch-by-patch calculations.

Although the method is applicable in higher dimensions, we have exercised it on the geometrically exact theory of Kirchhoff–Love thin-shells. Our work significantly extends the applicability of meshfree methods to thin-shell analysis. Previous methods were limited to very simple surfaces admitting a single parametric space. The proposed method is very robust and general and can deal very easily with shells of very complex geometry and topology. Furthermore, we have shown that it is very accurate and competitive with state-of-the-art mesh-based methods such as subdivision finite elements. Interestingly, although the local *max-ent* approximants are only linearly reproducing, we obtain excellent results for the fourth-order partial differential equations of thin-shells. We have observed the same behavior in the numerical approximation of a fourth-order phase-field model for biomembranes [74]. This suggests further mathematical analysis of the method [75]. The proposed method can be easily enhanced to account for internal connections or non-manifold shells [72, 76]. We are also working on a boundary representation avoiding ghost nodes, by describing the boundary curves by B-spline or NURBS curves [77].

The general methodology proposed here can be applied to many problems in science and engineering. We are currently exploiting it for the quantitative analysis of swimming strokes in micro-organisms, the compact model reduction of dynamical systems whose near-invariant manifold is nonlinear, or the automatic detection of meaningful collective variables in biomolecular simulations.

APPENDIX A: CLOSEST-POINT PROJECTION ONTO THE MANIFOLD

We describe here how to perform the closest-point projection of a point $\boldsymbol{x} \in \mathbb{R}^D$ close to the manifold onto the manifold. To simplify the notation, we restrict ourselves to a given patch. If $\boldsymbol{x} \in \mathcal{M}^D$, this can be interpreted as 'inverting' the parametrization to find a point $\boldsymbol{\xi} \in \mathbb{R}^d$ such that $\boldsymbol{\varphi}(\boldsymbol{\xi}) = \boldsymbol{x}$. For this purpose, we minimize the cost function $f(\boldsymbol{\xi}) = (1/2)|\boldsymbol{\varphi}(\boldsymbol{\xi}) - \boldsymbol{x}|^2$. We solve this nonlinear optimization problem with Newton's method

$$\boldsymbol{\xi}^{i+1} = \boldsymbol{\xi}^i - \boldsymbol{J}^{-1}(\boldsymbol{\xi}^i)\, \boldsymbol{r}(\boldsymbol{\xi}^i),$$

where $\boldsymbol{r}(\boldsymbol{\xi}) = \nabla f(\boldsymbol{\xi})$ is the gradient and $\boldsymbol{J}(\boldsymbol{\xi}) = Hf(\boldsymbol{\xi})$ is the Hessian of the cost function. The gradient of the cost function is given as

$$\nabla f(\boldsymbol{\xi}) = D\boldsymbol{\varphi}(\boldsymbol{\xi}) \cdot (\boldsymbol{\varphi}(\boldsymbol{\xi}) - \boldsymbol{x}),$$

where

$$D\boldsymbol{\varphi}(\boldsymbol{\xi}) = \sum_a \nabla p_a(\boldsymbol{\xi}) \otimes \boldsymbol{P}_a.$$

DOI: 10.1002/nme

The Hessian of the cost function can be written as

$$Hf(\boldsymbol{\xi}) = D^2\boldsymbol{\varphi}(\boldsymbol{\xi}) \cdot (\boldsymbol{\varphi}(\boldsymbol{\xi}) - \boldsymbol{x}) + D\boldsymbol{\varphi}(\boldsymbol{\xi}) \otimes D\boldsymbol{\varphi}(\boldsymbol{\xi}),$$

where

$$D^2\boldsymbol{\varphi}(\boldsymbol{\xi}) = \sum_a Hp_a(\boldsymbol{\xi}) \otimes \boldsymbol{P}_a.$$

As a starting point for Newton's method, we select $\boldsymbol{\xi}^0 = \sum_a w_a^P(\boldsymbol{x})\ \boldsymbol{\xi}_a$; here, w_a^P are the local linear weights from the kth nearest neighbors of $\boldsymbol{x}$ belonging to P. The local weights are computed in the spirit of LLE [50, 51]. This procedure is very robust and fast. We denote the solution of the minimization problem, the preimage of the closest point projection of a point in the high-dimensional space, by $\boldsymbol{\xi} = \boldsymbol{\varphi}^{-1}(\boldsymbol{\pi}(\boldsymbol{x}))$.

APPENDIX B: CONTROL POINTS BY A LEAST-SQUARE FIT TO THE SAMPLED MANIFOLD

Let $X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_K\},\ K > N$ be a good sampling of the manifold $\mathcal{M} \in \mathbb{R}^D$. Let us assume we have a reasonably good set of control points, for example, lying on the manifold, which we use to define the embedding and the *max-ent* basis functions. We wish to find the control points $\boldsymbol{P}_a,\ a = 1, \ldots, N$ such that the numerical surface best fits the data in an L_2 sense. For this purpose, we minimize the cost function

$$g(\boldsymbol{P}) = \frac{1}{2} \sum_{i=1}^{K} |I(\boldsymbol{x}_i) - \boldsymbol{x}_i|^2,$$

where $\boldsymbol{P}$ denotes a vector with the coordinates of all the control points. The reconstruction operator can be rewritten as

$$I(\boldsymbol{x}) = \sum_{\kappa \in \mathcal{N}_x} \psi_\kappa(\boldsymbol{x}) \sum_{a \in \mathcal{J}_\kappa} p_a(\boldsymbol{\xi}_\kappa(\boldsymbol{x}))\, \boldsymbol{P}_a = \boldsymbol{M}\boldsymbol{P},$$

where $\boldsymbol{M} \in \mathbb{R}^{K \times N}$ is a sparse matrix. The unique solution to this linear least-squares problem follows from the sparse linear system of equations $\boldsymbol{M}^{\mathrm{T}}\boldsymbol{M}\boldsymbol{P} = \boldsymbol{M}^{\mathrm{T}}\boldsymbol{X}$, where $\boldsymbol{X}$ is a vector collecting all the coordinates of the sampling points.

APPENDIX C: OUT-OF-BALANCE FORCES AND TANGENT STIFFNESS MATRIX

We provide here expressions for the gradient and the Hessian of the potential energy. With a view on the implementation, we resort to Voigt's notation for symmetric tensors. To keep the notation clean, depending on the context, we ignore the subscript indicating the patch number, that is, for instance, we denote $\boldsymbol{\varphi}_{0\kappa}$, a reference configuration of the middle surface in the κth patch simply by $\boldsymbol{\varphi}_0$. By the chain rule, the gradient of the discrete internal energy is

$$\frac{\partial \Pi_{\mathrm{int}}^h}{\partial \boldsymbol{P}_a} = \sum_{\kappa=1}^{L} \int_{\mathscr{A}_\kappa} \left[\left(\boldsymbol{n}^{\mathrm{T}} \frac{\partial \boldsymbol{\varepsilon}}{\partial \boldsymbol{P}_a} + \boldsymbol{m}^{\mathrm{T}} \frac{\partial \boldsymbol{\rho}}{\partial \boldsymbol{P}_a} \right) \bar{j}_0 \right]_\kappa (\psi_\kappa \circ \boldsymbol{\varphi}_0)\, \mathrm{d}\xi^1 \mathrm{d}\xi^2,$$

where the subindex κ means that the expression between the brackets is computed with the local parametrization of the κth patch.

The derivatives of the membrane and bending strain tensors with respect to the ath control point can be expressed in terms of the nonlinear membrane and bending strain–displacement matrices, $\boldsymbol{M}^a$ and $\boldsymbol{B}^a$, respectively, as

$$\frac{\partial \boldsymbol{\varepsilon}}{\partial \boldsymbol{P}_a} = \boldsymbol{M}^a, \quad \frac{\partial \boldsymbol{\rho}}{\partial \boldsymbol{P}_a} = \boldsymbol{B}^a.$$

DOI: 10.1002/nme

We express the strain–displacement matrices by introducing auxiliary vectors

$$M^a_{ij} = \boldsymbol{M}^a_i \cdot \boldsymbol{e}_j \quad \text{and} \quad B^a_{ij} = \boldsymbol{B}^a_i \cdot \boldsymbol{e}_j,$$

which can be written as

$$\begin{aligned}
\boldsymbol{M}^a_\alpha &= p_{a,\alpha}\, \boldsymbol{\varphi}_{,\alpha},\\
\boldsymbol{M}^a_3 &= p_{a,2}\, \boldsymbol{\varphi}_{,1} + p_{a,1}\, \boldsymbol{\varphi}_{,2},\\
\boldsymbol{B}^a_\alpha &= -p_{a,\alpha\alpha}\, \boldsymbol{t}\\
&\quad + \bar{j}^{-1} \left[(\boldsymbol{\varphi}_{,\alpha\alpha} \times \boldsymbol{\varphi}_{,2}) p_{a,1} + (\boldsymbol{\varphi}_{,1} \times \boldsymbol{\varphi}_{,\alpha\alpha}) p_{a,2} \right]\\
&\quad + \bar{j}^{-1} (\boldsymbol{t} \cdot \boldsymbol{\varphi}_{,\alpha\alpha}) \left[(\boldsymbol{\varphi}_{,2} \times \boldsymbol{t}) p_{a,1} + (\boldsymbol{t} \times \boldsymbol{\varphi}_{,1}) p_{a,2} \right],\\
\boldsymbol{B}^a_3 &= -2\, p_{a,12}\, \boldsymbol{t}\\
&\quad + 2\, \bar{j}^{-1} \left[(\boldsymbol{\varphi}_{,12} \times \boldsymbol{\varphi}_{,2}) p_{a,1} + (\boldsymbol{\varphi}_{,1} \times \boldsymbol{\varphi}_{,12}) p_{a,2} \right]\\
&\quad + 2\, \bar{j}^{-1} (\boldsymbol{t} \cdot \boldsymbol{\varphi}_{,12}) \left[(\boldsymbol{\varphi}_{,2} \times \boldsymbol{t}) p_{a,1} + (\boldsymbol{t} \times \boldsymbol{\varphi}_{,1}) p_{a,2} \right].
\end{aligned}$$

By $\boldsymbol{e}_j$, we denote the canonical basis vectors of $\mathbb{R}^3$. Note that repeated indices in the expressions for $\boldsymbol{M}^a_\alpha$ and $\boldsymbol{B}^a_\alpha$ do not imply summation.

For the tangent stiffness matrix, ignoring follower loads, we compute the second-order partial derivatives of the internal potential energy with respect to the control point positions as

$$\frac{\partial^2 \Pi^h_{\text{int}}}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b} = \frac{1}{2} \sum_{\kappa=1}^{L} \int_{\mathscr{A}_\kappa} \left[\frac{\partial^2 (\boldsymbol{\varepsilon} \cdot \boldsymbol{n} + \boldsymbol{\rho} \cdot \boldsymbol{m})}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b} \bar{j}_0 \right]_\kappa (\psi_\kappa \circ \boldsymbol{\varphi}_0)\, \mathrm{d}\xi^1 \mathrm{d}\xi^2,$$

where

$$\frac{\partial^2 (\boldsymbol{\varepsilon} \cdot \boldsymbol{n} + \boldsymbol{\rho} \cdot \boldsymbol{m})}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b} = \boldsymbol{n}^{\mathrm{T}} \frac{\partial^2 \boldsymbol{\varepsilon}}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b} + h \boldsymbol{M}^{a\mathrm{T}} \boldsymbol{C}\, \boldsymbol{M}^b + \boldsymbol{m}^{\mathrm{T}} \frac{\partial^2 \boldsymbol{\rho}}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b} + \frac{h^3}{12} \boldsymbol{B}^{a\mathrm{T}} \boldsymbol{C}\, \boldsymbol{B}^b,$$

and $\boldsymbol{C}$ denotes the Voigt representation of $C^{\alpha\beta\gamma\delta}$. The second derivatives of the membrane and bending strain tensors can be computed as

$$\frac{\partial^2 \boldsymbol{\varepsilon}}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b} = \begin{pmatrix} p_{a,1}\ p_{b,1} \\ p_{a,2}\ p_{b,2} \\ p_{a,1}\ p_{b,2} + p_{a,2}\ p_{b,1} \end{pmatrix} \otimes \boldsymbol{I},$$

and

$$\begin{aligned}
\boldsymbol{m}^{\mathrm{T}} \frac{\partial^2 \boldsymbol{\rho}}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b} = &-\bar{j}^{-1} \left[\frac{\partial \bar{j}}{\partial \boldsymbol{P}_a} \otimes \boldsymbol{f}^b_\rho + \boldsymbol{f}^a_\rho \otimes \frac{\partial \bar{j}}{\partial \boldsymbol{P}_b} \right] + \bar{j}^{-1} \left(\boldsymbol{m}^{\mathrm{T}} H \boldsymbol{\varphi}\, \boldsymbol{t} \right) \frac{\partial^2 \bar{j}}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b}\\
&- \bar{j}^{-1} (\boldsymbol{m} \cdot H p_a) [\boldsymbol{w}_b]_\times - \bar{j}^{-1} (\boldsymbol{m} \cdot H p_b) [\boldsymbol{w}_a]^{\mathrm{T}}_\times\\
&- \bar{j}^{-1} (p_{a,2} p_{b,1} - p_{a,1} p_{b,2}) \left[(\boldsymbol{m}^{\mathrm{T}} H \boldsymbol{\varphi})^{\mathrm{T}} \right]_\times.
\end{aligned}$$

In the preceding equation, $\boldsymbol{f}^a_\rho = [\boldsymbol{m}^{\mathrm{T}} \boldsymbol{B}^a]^{\mathrm{T}}$, $[\boldsymbol{v}]_\times$ denotes the skew-symmetric matrix

$$[\boldsymbol{v}]_\times \overset{\text{def}}{=} \begin{pmatrix} 0 & -\boldsymbol{v} \cdot \boldsymbol{e}_3 & \boldsymbol{v} \cdot \boldsymbol{e}_2 \\ \boldsymbol{v} \cdot \boldsymbol{e}_3 & 0 & -\boldsymbol{v} \cdot \boldsymbol{e}_1 \\ -\boldsymbol{v} \cdot \boldsymbol{e}_2 & \boldsymbol{v} \cdot \boldsymbol{e}_1 & 0 \end{pmatrix},$$

and we introduce

$$\boldsymbol{w}_a = -p_{a,1}\, \boldsymbol{\varphi}_{,2} + p_{a,2}\, \boldsymbol{\varphi}_{,1}$$

DOI: 10.1002/nme

and

$$H\boldsymbol{\varphi} = \begin{pmatrix} \boldsymbol{\varphi}_{,11}\cdot\boldsymbol{e}_1 & \boldsymbol{\varphi}_{,11}\cdot\boldsymbol{e}_2 & \boldsymbol{\varphi}_{,11}\cdot\boldsymbol{e}_3 \\ \boldsymbol{\varphi}_{,22}\cdot\boldsymbol{e}_1 & \boldsymbol{\varphi}_{,22}\cdot\boldsymbol{e}_2 & \boldsymbol{\varphi}_{,22}\cdot\boldsymbol{e}_3 \\ 2\boldsymbol{\varphi}_{,12}\cdot\boldsymbol{e}_1 & 2\boldsymbol{\varphi}_{,12}\cdot\boldsymbol{e}_2 & 2\boldsymbol{\varphi}_{,12}\cdot\boldsymbol{e}_3 \end{pmatrix} \quad Hp_a = \begin{pmatrix} p_{a,11} \\ p_{a,22} \\ 2p_{a,12} \end{pmatrix},$$

to keep the notation compact.

The first-order and second-order partial derivatives of the Jacobian are

$$\frac{\partial \bar{j}}{\partial \boldsymbol{P}_a} = p_{a,1}(\boldsymbol{\varphi}_{,2}\times\boldsymbol{t}) + p_{a,2}(\boldsymbol{t}\times\boldsymbol{\varphi}_{,1})$$

and

$$\frac{\partial^2 \bar{j}}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b} = -\bar{j}^{-1}\frac{\partial \bar{j}}{\partial \boldsymbol{P}_a}\otimes\frac{\partial \bar{j}}{\partial \boldsymbol{P}_b} + \bar{j}^{-1}[\boldsymbol{w}_a]_\times^{\mathrm{T}}[\boldsymbol{w}_b]_\times + (-p_{a,1}\ p_{b,2} + p_{a,2}\ p_{b,1})[\boldsymbol{t}]_\times.$$

To conclude, we give the explicit first and second derivates of the normal contracted with an auxiliary vector, $\boldsymbol{v}\in\mathbb{R}^3$:

$$\frac{\partial \boldsymbol{t}}{\partial \boldsymbol{P}_a} = -\bar{j}^{-1}\left(\boldsymbol{t}\otimes\frac{\partial \bar{j}}{\partial \boldsymbol{P}_a} - [\boldsymbol{w}_a]_\times\right),$$

$$\left(\boldsymbol{v}^{\mathrm{T}}\frac{\partial \boldsymbol{t}}{\partial \boldsymbol{P}_a}\right)^{\mathrm{T}} = -\bar{j}^{-1}\left[(\boldsymbol{v}\cdot\boldsymbol{t})\frac{\partial \bar{j}}{\partial \boldsymbol{P}_a} - \left(p_{a,1}(\boldsymbol{\varphi}_{,2}\times\boldsymbol{v}) + p_{a,2}(\boldsymbol{v}\times\boldsymbol{\varphi}_{,1})\right)\right],$$

and

$$\boldsymbol{v}^{\mathrm{T}}\frac{\partial^2 \boldsymbol{t}}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b} = -\bar{j}^{-1}\left[(\boldsymbol{v}\cdot\boldsymbol{t})\frac{\partial^2 \bar{j}}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b} + \frac{\partial \bar{j}}{\partial \boldsymbol{P}_a}\otimes\left(\boldsymbol{v}^{\mathrm{T}}\frac{\partial \boldsymbol{t}}{\partial \boldsymbol{P}_b}\right)^{\mathrm{T}} + \left(\boldsymbol{v}^{\mathrm{T}}\frac{\partial \boldsymbol{t}}{\partial \boldsymbol{P}_a}\right)^{\mathrm{T}}\otimes\frac{\partial \bar{j}}{\partial \boldsymbol{P}_b}\right] + \bar{j}^{-1}(-p_{a,1}\ p_{b,2} + p_{a,2}\ p_{b,1})[\boldsymbol{v}]_\times.$$

These expressions are needed, for example, to impose essential boundary conditions of rotation (Appendix D).

APPENDIX D: ESSENTIAL BOUNDARY CONDITIONS

We describe here the numerical constraints needed to impose the essential boundary conditions, for both displacements and rotations. We describe the variational formulation with Lagrange multipliers and the matrices needed in the augmented Lagrangian scheme.

Let us consider first the integral of a function f over the lateral boundary surface $\partial\mathscr{S}_0$ of a thin-shell object $\mathscr{S}_0$ ($\partial\mathscr{S}_0$ excludes the body boundary surfaces parallel to the middle surface). Assuming the function does not change through the thickness, we have

$$\int_{\partial\mathscr{S}_0} f\,\mathrm{d}S_0 = \int_{\partial\Omega_0} f\left(\int_{-\frac{h}{2}}^{\frac{h}{2}} \frac{\left|\frac{\partial\boldsymbol{\Phi}_0}{\partial\xi}\times\frac{\partial\boldsymbol{\Phi}_0}{\partial t}\right|}{\left|\boldsymbol{\varphi}_{0,t}\right|}\mathrm{d}\xi\right)\mathrm{d}\ell_0,$$

where t is a tangent coordinate along the boundary curve $\partial\mathscr{A}$. By introducing $\partial\boldsymbol{\Phi}_0/\partial t = \boldsymbol{\varphi}_{0,t} + \xi\,\boldsymbol{t}_{0,t}$, we obtain

$$\int_{-\frac{h}{2}}^{\frac{h}{2}} \frac{\left|\boldsymbol{t}_0\times\frac{\partial\boldsymbol{\Phi}_0}{\partial t}\right|}{\left|\boldsymbol{\varphi}_{0,t}\right|}\mathrm{d}\xi = h\frac{\left|\boldsymbol{t}_0\times\boldsymbol{\varphi}_{0,t}\right|}{\left|\boldsymbol{\varphi}_{0,t}\right|}.$$

DOI: 10.1002/nme

With the previous expressions and the PU, the integral of a function f on the boundary surface $\partial\mathscr{S}_0$ becomes

$$\int_{\partial\mathscr{S}_0} f \, \mathrm{d}S_0 = \int_{\partial\Omega_0} h \, f \, \frac{\left|\boldsymbol{t}_0 \times \boldsymbol{\varphi}_{0,t}\right|}{\left|\boldsymbol{\varphi}_{0,t}\right|} \, \mathrm{d}\ell_0 = \sum_{\kappa=1}^{L} \int_{\partial\mathscr{A}_\kappa} \left[h \, (f \circ \boldsymbol{\varphi}_0) \left|\boldsymbol{t}_0 \times \boldsymbol{\varphi}_{0,t}\right|\right]_\kappa (\psi_\kappa \circ \boldsymbol{\varphi}_0) \, \mathrm{d}\ell_\xi.$$

Here, subindex κ means that the expression between the brackets is computed with the local parametrization of the κth patch.

Displacement constraints on a curve

Let $\boldsymbol{\lambda}_u$ be the Lagrange multipliers field associated to the displacement constraints $\boldsymbol{\varphi} = \bar{\boldsymbol{\varphi}}$ on $\partial\Omega^u$. We discretize the Lagrange multipliers as $\boldsymbol{\lambda}_u = \sum_i N_i(\xi) \, \boldsymbol{\Lambda}_i^u$, where N_i are the standard piecewise linear basis functions defined from the boundary nodes. With the PU, the displacement constraints can be expressed variationally as

$$0 = \int_{\Omega_0^u} h \, \boldsymbol{\lambda}_u \cdot (\boldsymbol{\varphi} - \bar{\boldsymbol{\varphi}}) \frac{\left|\boldsymbol{t}_0 \times \boldsymbol{\varphi}_{0,t}\right|}{\left|\boldsymbol{\varphi}_{0,t}\right|} \, \mathrm{d}\ell_0 = \sum_{\kappa=1}^{L} \int_{\partial\mathscr{A}_\kappa^u} \left\{h \, \boldsymbol{\lambda}_u \cdot (\boldsymbol{\varphi} - \bar{\boldsymbol{\varphi}}) \, \left|\boldsymbol{t}_0 \times \boldsymbol{\varphi}_{0,t}\right|\right\}_\kappa (\psi_\kappa \circ \boldsymbol{\varphi}_0) \, \mathrm{d}\ell_\xi,$$

where $\partial\mathscr{A}_\kappa^u = \boldsymbol{\varphi}_{0\kappa}^{-1}(\partial\Omega^u \cap \mathrm{supp}(\psi_\kappa))$, for all Lagrange multipliers. Recalling their discretization, we can write the constraint in matrix form as $\boldsymbol{C}_u(P) = \boldsymbol{0}$, where

$$\boldsymbol{C}_i^u(P) = \sum_{\kappa=1}^{L} \int_{\partial\mathscr{A}_\kappa^u} \left\{h \, N_i \, (\boldsymbol{\varphi} - \bar{\boldsymbol{\varphi}}) \, \left|\boldsymbol{t}_0 \times \boldsymbol{\varphi}_{0,t}\right|\right\}_\kappa (\psi_\kappa \circ \boldsymbol{\varphi}_0) \, \mathrm{d}\ell_\xi.$$

These constraints are linear, with

$$\frac{\partial \boldsymbol{C}_i^u}{\partial \boldsymbol{P}_a} = \left(\sum_{\kappa=1}^{L} \int_{\partial\mathscr{A}_\kappa^u} \left\{h \, N_i \, p_a \, \left|\boldsymbol{t}_0 \times \boldsymbol{\varphi}_{0,t}\right|\right\}_\kappa (\psi_\kappa \circ \boldsymbol{\varphi}_0) \, \mathrm{d}\ell_\xi \right) \boldsymbol{I},$$

and $\partial^2 \boldsymbol{C}_i^u / \partial \boldsymbol{P}_a \partial \boldsymbol{P}_b = 0$.

Rotation constraints on a curve

Let $\boldsymbol{\tau}_0 = \boldsymbol{\varphi}_{0,t} / \left|\boldsymbol{\varphi}_{0,t}\right|$ be a unit vector tangent to the boundary curve of the middle surface Ω_0 satisfying $\boldsymbol{t}_0 \cdot \boldsymbol{\tau}_0 = 0$. The rotation boundary conditions take the form $\boldsymbol{\nu}_0 \cdot \boldsymbol{t} = \bar{g}_\theta$ on $\partial\Omega^\theta$, where $\boldsymbol{\nu}_0 = \boldsymbol{t}_0 \times \boldsymbol{\tau}_0$ is the outward tangent vector to the boundary curve. With λ_θ denoting the Lagrange multiplier field associated with this constraint, the variational statement of the constraint is

$$0 = \int_{\partial\Omega_0^\theta} h \, \lambda_\theta \, (\boldsymbol{\nu}_0 \cdot \boldsymbol{t} - \bar{g}_\theta) \frac{\left|\boldsymbol{t}_0 \times \boldsymbol{\varphi}_{0,t}\right|}{\left|\boldsymbol{\varphi}_{0,t}\right|} \, \mathrm{d}\ell_0 = \sum_{\kappa=1}^{L} \int_{\partial\mathscr{A}_\kappa^\theta} \left\{h \, \lambda_\theta \, (\boldsymbol{\nu}_0 \cdot \boldsymbol{t} - \bar{g}_\theta) \, \left|\boldsymbol{t}_0 \times \boldsymbol{\varphi}_{0,t}\right|\right\}_\kappa (\psi_\kappa \circ \boldsymbol{\varphi}_0) \, \mathrm{d}\ell_\xi,$$

for all Lagrange multipliers. With the discrete representation of the Lagrange multipliers $\lambda_\theta = \sum_i N_i(\xi) \, \Lambda_i^\theta$, the discrete constraints become

$$0 = C_i^\theta(P) = \sum_{\kappa=1}^{L} \int_{\partial\mathscr{A}_\kappa^\theta} \left\{h \, N_i \, (\boldsymbol{\nu}_0 \cdot \boldsymbol{t} - \bar{g}_\theta) \, \left|\boldsymbol{t}_0 \times \boldsymbol{\varphi}_{0,t}\right|\right\}_\kappa (\psi_\kappa \circ \boldsymbol{\varphi}_0) \, \mathrm{d}\ell_\xi.$$

DOI: 10.1002/nme

These constraints are nonlinear. For the augmented Lagrangian implementation, we need

$$\frac{\partial C_i^\theta}{\partial \boldsymbol{P}_a} = \sum_{\kappa=1}^{L} \int_{\partial \mathscr{A}_\kappa^\theta} \left\{ h\ N_i\ \left(\boldsymbol{v}_0^{\mathrm{T}} \frac{\partial \boldsymbol{t}}{\partial \boldsymbol{P}_a} \right)^{\mathrm{T}} |\boldsymbol{t}_0 \times \boldsymbol{\varphi}_{0,t}| \right\}_\kappa (\psi_\kappa \circ \boldsymbol{\varphi}_0)\ \mathrm{d}\ell_\xi$$

and

$$\frac{\partial^2 C_i^\theta}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b} = \sum_{\kappa=1}^{L} \int_{\partial \mathscr{A}_\kappa^\theta} \left\{ h\ N_i\ \left(\boldsymbol{v}_0^{\mathrm{T}} \frac{\partial^2 \boldsymbol{t}}{\partial \boldsymbol{P}_a \partial \boldsymbol{P}_b} \right) |\boldsymbol{t}_0 \times \boldsymbol{\varphi}_{0,t}| \right\}_\kappa (\psi_\kappa \circ \boldsymbol{\varphi}_0)\ \mathrm{d}\ell_\xi.$$

Imposed displacement on a point

Suppose we want to constrain a point $\boldsymbol{x}_0 \in \Omega_0$ to be at location $\boldsymbol{x}_1$ in the deformed configuration. By defining $\boldsymbol{\xi}_0 = \boldsymbol{\varphi}_0^{-1}(\boldsymbol{x}_0)$, the preimage of this point by the reference configuration of the middle surface, we can write this constraint as

$$\boldsymbol{0} = \boldsymbol{C}^p(P) = \boldsymbol{\varphi}(\boldsymbol{\xi}_0) - \boldsymbol{x}_1.$$

APPENDIX E: AUGMENTED LAGRANGIAN SOLUTION METHOD

The equilibrium solutions of the nonlinear constrained minimization problem described in Section 4.4 are stationary points of the Lagrangian

$$\mathcal{L}(P, \boldsymbol{\Lambda}) = \Pi^h(P) - \boldsymbol{\Lambda}^{\mathrm{T}} \boldsymbol{C}(P),$$

where $\boldsymbol{C}$ collects all the discrete constraints of the previous sections and $\boldsymbol{\Lambda}$ collects the corresponding Lagrange multipliers (reaction forces and torques). The Kuhn–Tucker optimality conditions, $\partial_P \mathcal{L} = 0$, $\partial_{\boldsymbol{\Lambda}} \mathcal{L} = 0$, may be solved with Newton's method, yet this approach may lead to unstable equilibria and avoid physically relevant buckled stable solutions.

A robust strategy that guarantees stable equilibria is based on the augmented Lagrangian method, which combines the standard Lagrangian with penalties. This method retains the exactness of the Lagrange multipliers method and the minimization principle of penalty methods. The minimization is performed iteratively on the control points only for frozen Lagrange multipliers, which are updated explicitly (see [78, 79] for further details). The augmented Lagrangian is

$$\mathcal{L}_A(P, \boldsymbol{\Lambda}) = \Pi^h(P) - \boldsymbol{\Lambda}^{\mathrm{T}} \boldsymbol{C}(P) + \frac{1}{2\mu} \boldsymbol{C}(P)^{\mathrm{T}} \boldsymbol{C}(P),$$

where μ is the penalty parameter. We solve the problem in two stages. First, we find an approximate minimizer with a slow, robust method and a coarse tolerance. For this, in the augmented Lagrangian inner minimization loop, we adopt the limited-memory Broyden-Fletcher-Goldfarb-Shanno algorithm. Then, the minimizer is refined by resorting to Newton's method with line search.

ACKNOWLEDGEMENTS

We acknowledge the support of the European Research Council under the European Community's 7th Framework Programme (FP7/2007-2013)/ERC grant agreement no. 240487. MA acknowledges the support received through the prize 'ICREA Academia' for excellence in research, funded by the Generalitat de Catalunya.

REFERENCES

1. MacNeal R, Harder R. A proposed standard set of problems to test finite element accuracy. *Finite Element in Analysis and Design* 1985; **1**(1):3–20.
2. Bucalem M, Bathe J. Higher-order MITC general shell elements. *International Journal for Numerical Methods in Engineering* 1993; **36**(21):3729–3754.

DOI: 10.1002/nme

3. Simo J, Fox D. On a stress resultant geometrically exact shell model. Part I: formulation and optimal parametrization. *Computer Methods in Applied Mechanics and Engineering* 1989; **72**:267–304.
4. Krysl P, Belytschko T. Analysis of thin shells by the element-free Galerkin method. *International Journal of Solids and Structures* 1996; **33**(20–22):3057–3078.
5. Cirak F, Ortiz M, Schröder P. Subdivision surfaces: a new paradigm for thin-shell finite-element analysis. *International Journal for Numerical Methods in Engineering* 2000; **47**(12):2039–2072.
6. Cirak F, Ortiz M. Fully $\boldsymbol{C}^1$-conforming subdivision elements for finite deformation thin-shell analysis. *International Journal for Numerical Methods in Engineering* 2001; **51**(7):813–833.
7. Hughes T, Cottrell J, Bazilevs Y. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering* 2005; **194**:4135–4195.
8. Cottrell J, Hughes T, Bazilevs Y. *Isogeometric Analysis: Toward Integration of CAD and FEA*. John Wiley & Sons, Ltd.: Hoboken, NJ, 2009.
9. Cottrell J, Reali A, Bazilevs Y, Hughes T. Isogeometric analysis of structural vibrations. *Computer Methods in Applied Mechanics and Engineering* 2006; **195**(41–43):5257–5296.
10. Kiendl J, Bletzinger KU, Linhard J, Wüchner R. Isogeometric shell analysis with Kirchhoff–Love elements. *Computer Methods in Applied Mechanics and Engineering* 2009; **198**(49–52):3902–3914.
11. Bazilevs Y, Calo V, Cottrell J, Evans J, Hughes T, Lipton S, Scott M, Sederberg T. Isogeometric analysis using T-splines. *Computer Methods in Applied Mechanics and Engineering* 2010; **199**(5–8):229–263.
12. Engel G, Garikipati K, Hughes T, Larson M, Mazzei L, Taylor R. Continuous/discontinuous finite element approximations of fourth-order elliptic problems in structural and continuum mechanics with applications to thin beams and plates, and strain gradient elasticity. *Computer Methods in Applied Mechanics and Engineering* 2002; **191**:3669–3750.
13. Wells G, Dung N. A $\boldsymbol{C}^0$ discontinuous Galerkin formulation for Kirchhoff plates. *Computer Methods in Applied Mechanics and Engineering* 2007; **196**(35–36):3370–3380.
14. Wells G, Dung N. Geometrically nonlinear formulation for thin shells without rotation degrees of freedom. *Computer Methods in Applied Mechanics and Engineering* 2008; **197**(35–36):3370–3380.
15. Noels L, Radovitzky R. A new discontinuous Galerkin method for Kirchhoff–Love shells. *Computer Methods in Applied Mechanics and Engineering* 2008; **197**(33–40):2901–2929.
16. Noels L. A discontinuous Galerkin formulation of non-linear Kirchhoff–Love shells. *International Journal for Numerical Methods in Engineering* 2009; **78**(3):296–323.
17. Hoppe H, DeRose T, Duchamp T, McDonald J, Stuetzle W. Surface reconstruction from unorganized points. In *SIGGRAPH '92: Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques*. ACM: New York, NY, 1992; 71–78.
18. Alexa M, Behr J, Cohen-Or D, Fleishman S, Levin D, Silva C. Point set surfaces. In *VIS '01: Proceedings of the Conference on Visualization '01*. IEEE Computer Society: Washington, DC, 2001; 21–28.
19. Pauly M. Point primitives for interactive modeling and processing of 3D geometry. *PhD Thesis*, Federal Institute of Technology (ETH) of Zurich, 2003.
20. Levin D. Mesh-independent surface interpolation. In *Geometric Modeling for Scientific Visualization*, Brunnett G, Hamann B, Mueller H, Linsen L (eds). Springer-Verlag: Heidelberg, 2003; 37–49.
21. Ohtake Y, Belyaev A, Alexa M, Turk G, Seidel H. Multi-level partition of unity implicits. *ACM Transactions on Graphics (Proc. SIGGRAPH 2003)* 2003; **22**:463–470.
22. Alexa M, Behr J, Cohen-Or D, Fleishman S, Levin D, Silva C. Computing and rendering point set surfaces. *Transactions on Visualization and Computer Graphics* 2003; **9**(1):3–15.
23. Amenta N, Kil Y. Defining point-set surfaces. *ACM Transactions on Graphics* 2004; **23**(3):264–270.
24. Alexa M, Gross M, Pauly M, Pfister H, Stamminger M, Zwicker M. Point-based computer graphics. *SIGGRAPH* 2004 Course Notes 2004.
25. Noguchi H, Kawashima T, Miyamura T. Element free analyses of shell and spatial structures. *International Journal for Numerical Methods in Engineering* 2000; **47**(6):1215–1240.
26. Chen J, Wang D. A constrained reproducing kernel particle formulation for shear deformable shell in Cartesian coordinates. *International Journal for Numerical Methods in Engineering* 2006; **68**(2):151–172.
27. Rabczuk T, Areias P, Belytschko T. A meshfree thin shell method for non-linear dynamic fracture. *International Journal for Numerical Methods in Engineering* 2007; **72**(5):524–548.
28. Millán D, Rosolen A, Arroyo M. Thin shell analysis from scattered points with maximum-entropy approximants. *International Journal for Numerical Methods in Engineering* 2011; **85**(6):723–751.
29. Belytschko T, Stolarski H, Liu W, Carpenter N, Ong J. Stress projection for membrane and shear locking in shell finite-elements. *Computer Methods in Applied Mechanics and Engineering* 1985; **51**:221–258.
30. Arroyo M, Ortiz M. Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods. *International Journal for Numerical Methods in Engineering* 2006; **65**(13):2167–2202.
31. Pearson K. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine* 1901; **2**(6):559–572.
32. Hotelling H. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology* 1933; **24**(7):498–520.
33. Karhunen K. Zur spektraltheorie stochastischer prozesse. *Annales Academiae Scientiarum Fennicae* 1946; **34**:1–7.
34. Loève M. *Probability Theory.* University series in higher mathematics, Van Nostrand: New Jersey, 1955.

DOI: 10.1002/nme

35. Lorenz E. Empirical orthogonal functions and statistical weather prediction. *Statistical Forecasting Project, Scientific Report 1*, MIT, Department of Meteorology, Cambridge, MA, 1956.
36. Lumley JL. The structure of inhomogeneous turbulent flows. In *Atmospheric Turbulence and Radio Propagation*, Yaglom AM, Tatarski VI (eds). Nauka: Moscow, 1967; 166–178.
37. Jolliffe IT. *Principal Component Analysis*, 2nd edn, Springer Series in Statistics, Springer: New York, NY, 2002.
38. Jain AK, Duin R, Mao J. Statistical pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2000; **22**(1):4–37.
39. Zhang Z, Zha H. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM Journal of Scientific Computing* 2005; **26**(1):313–338.
40. Lall S, Krysl P, Marsden J. Structure-preserving model reduction for mechanical systems. *Physica D* 2003; **184**:304–318.
41. Niroomandi S, Alfaro I, Cueto E, Chinesta F. Model order reduction for hyperelastic materials. *International Journal for Numerical Methods in Engineering* 2010; **81**(9):1180–1206.
42. Torgerson W. Multidimensional scaling: I. Theory and method. *Psychometrika* 1952; **17**:401–419.
43. Cox TF, Cox MAA. *Multidimensional Scaling*, 2nd edn. Chapman & Hall: Boca Raton, FL, 2001.
44. Lee J, Verleysen M. *Nonlinear Dimensionality Reduction*, Information Science and Statistics. Springer: New York, NY, 2007.
45. Sammon JW. A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers* 1969; **18**:401–409.
46. Hérault J, Oliva A, Guérin-Dugué A. Scene categorisation by curvilinear component analysis of low frequency spectra. *Proceedings of the 5th European Symposium on Artificial Neural Networks*, Bruges, Belgium, 1997; 91–96.
47. Hérault J, Jausions-Picaud C, Guérin-Dugué A. Curvilinear component analysis for high-dimensional data representation: I. Theoretical aspects and practical use in the presence of noise. In *Engineering Applications of Bio-Inspired Artificial Neural Networks*, Vol. 1607, Mira J, Sánchez-Andrés J (eds), Lecture Notes in Computer Science. Springer: Berlin/Heidelberg, 1999; 625–634.
48. van der Maaten LJP, Postma EO, van den Herik HJ. Dimensionality reduction: a comparative review. *Technical Report TiCC-TR 2009-005*, Tilburg University, 2009.
49. Tenenbaum J, de Silva V, Langford J. A global geometric framework for nonlinear dimensionality reduction. *Science* 2000; **290**(5500):2319–2323.
50. Roweis S, Saul L. Nonlinear dimensionality reduction by locally linear embedding. *Science* 2000; **290**(5500):2323–2326.
51. Roweis S, Saul L. Think globally, fit locally: unsupervised learning of low dimensional manifolds. *Journal of Machine Learning Research* 2003; **4**:119–155.
52. de Silva V, Tenenbaum J. Global versus local methods in nonlinear dimensionality reduction. In *Advances in Neural Information Processing Systems*, Vol. 15. MIT Press: Cambridge, MA, 2003; 705–712.
53. Choi H, Choi S. Robust kernel Isomap. *Pattern Recognition* 2007; **40**(3):853–862.
54. Donoho D, Grimes C. Hessian eigenmaps: locally linear embedding techniques for high-dimensional data. *Proceedings of the National Academy of Sciences* 2003; **100**(10):5591–5596.
55. Belkin M, Niyogi P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation* 2003; **15**:1373–1396.
56. Chang H, Yeung DY. Robust locally linear embedding. *Pattern Recognition* 2006; **39**(6):1053–1065.
57. Zhang Z, Wang J. MLLE: modified locally linear embedding using multiple weights. In *Advances in Neural Information Processing Systems*, Vol. 19, Schölkopf B, Platt J, Hoffman T (eds). MIT Press: Cambridge, MA, 2007; 1593–1600.
58. Gámez AJ, Zhou CS, Timmermann A, Kurths J. Nonlinear dimensionality reduction in climate data. *Nonlinear Processes in Geophysics* 2004; **11**:393–398.
59. Das P, Moll M, Stamati H, Kavraki L, Clementi C. Low-dimensional, free-energy landscapes of protein-folding reactions by nonlinear dimensionality reduction. *Proceedings of the National Academy of Sciences* 2006; **103**(26):9885–9890.
60. Brown WM, Martin S, Pollock SN, Coutsias EA, Watson JP. Algorithmic dimensionality reduction for molecular structure analysis. *The Journal of Chemical Physics* 2008; **129**(6):064118.
61. Vanderplas J, Connolly A. Reducing the dimensionality of data: locally linear embedding of Sloan galaxy spectra. *The Astronomical Journal* 2009; **138**(5):1365–1379.
62. do Carmo MP. *Differential Geometry of Curves and Surfaces*. Prentice-Hall, 1976.
63. de Ridder D, Duin R. Locally linear embedding for classification. *Technical Report PH-2002-0*, Department of Imaging Science & Technology, Delft University of Technology, Delft, 2002.
64. Grassberger P, Procaccia I. Measuring the strangeness of strange attractors. *Physica D* 1983; **9**(1–2):189–208.
65. Karypis G, Kumar V. METIS: Unstructured graph partitioning and sparse matrix ordering system, version 4.0, 1998. Available from: http://www.cs.umn.edu/~metis [Accessed on 2011].
66. Sha F, Saul L. Analysis and extension of spectral methods for nonlinear dimensionality reduction. In *Proceedings of the 22nd International Conference on Machine Learning*, ICML '05. ACM: New York, NY, 2005; 784–791.
67. Mount D, Arya S. ANN: a library for approximate nearest neighbor searching, version 1.1, 2010. Available from: http://www.cs.umd.edu/~mount/ANN [Accessed on 2011].
68. Simo J, Fox D, Rifai M. On a stress resultant geometrically exact shell model. Part II: the linear theory; computational aspects. *Computer Methods in Applied Mechanics and Engineering* 1989; **73**:53–92.

DOI: 10.1002/nme

69. Ciarlet PG. *Mathematical Elasticity: Theory of Shells*, Vol. III: North-Holland, 2000.
70. Sze KY, Liu XH, Lo SH. Popular benchmark problems for geometric nonlinear analysis of shells. *Journal Finite Elements in Analysis and Design* 2004; **40**:1551–1569.
71. Millán D. Geometrically nonlinear meshfree thin-shell analysis in the context of Kirchhoff–Love theory, 2011. Available from: http://www.youtube.com/RDanielMillan [Accessed on 2011].
72. Green S, Turkiyyah G. Second-order accurate constraint formulation for subdivision finite element simulation of thin shells. *International Journal for Numerical Methods in Engineering* 2004; **61**(3):380–405.
73. Turk G, Levoy M. Zippered polygon meshes from range images. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '94. ACM: New York, NY, 1994; 311–318.
74. Rosolen A, Millán D, Arroyo M. Second order convex maximum entropy approximants with applications to high order PDE. *International Journal for Numerical Methods in Engineering*, under revision, 2012.
75. Bompadre A, Perotti L, Cyron C, Ortiz M. Convergent meshfree approximation schemes of arbitrary order and smoothness. *Computer Methods in Applied Mechanics and Engineering* 2012:221–222:83–183.
76. Cirak F, Long Q. Subdivision shells with exact boundary control and non-manifold geometry. *International Journal for Numerical Methods in Engineering* 2011; **88**(9):897–923.
77. Rosolen A, Arroyo M. Blending isogeometric analysis and maximum entropy meshfree approximants. *In preparation*, 2012.
78. Nocedal J, Wright S. *Numerical Optimization*. Springer: New York, 1999.
79. Conn A, Gould N, Toint P. A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds. *SIAM Journal on Numerical Analysis* 1991; **28**(2):545–572.

DOI: 10.1002/nme

Appendix C

Paper # 3: "Nonlinear manifold learning for model reduction in finite elastodynamics"

Nonlinear manifold learning for model reduction in finite elastodynamics

Daniel Millan and Marino Arroyo*

Dept. Applied Mathematics 3, LaCàN,
Universitat Politècnica de Catalunya-BarcelonaTech, 08034 Barcelona, Spain

Abstract

Model reduction in computational mechanics is generally addressed with linear dimensionality reduction methods such as Principal Components Analysis (PCA). Hypothesizing that in many applications of interest the essential dynamics evolve on a nonlinear manifold, we explore here reduced order modeling based on nonlinear dimensionality reduction methods. Such methods are gaining popularity in diverse fields of science and technology, such as machine perception or molecular simulation. We consider finite deformation elastodynamics as a model problem, and identify the slow manifold by nonlinear dimensionality reduction methods applied to a database of snapshots. Contrary to linear dimensionality reduction, the smooth parametrization of the slow manifold needs special techniques, and we use local maximum entropy approximants. We then formulate the Lagragian mechanics on these data-based generalized coordinates, and develop variational time-integrators. Our proof-of-concept example shows that a few nonlinear collective variables provide similar accuracy to tens of PCA modes, suggesting that the proposed method may be very attractive in control or optimization applications. Furthermore, the reduced number of variables brings insight into the mechanics of the system under scrutiny. Our simulations also highlight the need of modeling the net effect of the disregarded degrees of freedom on the reduced dynamics at long times.

Keywords: Reduced order modeling, nonlinear dimensionality reduction, finite deformation elastodynamics, maximum entropy approximants, variational integrators

1. Introduction

The goal of model reduction is to process (simulate, control, optimize) a complex system in terms of a much simpler description. The premise is that, with a proper selection of the reduced model, the essential features of the original system can be preserved by a very light and efficient model. Most often, the reduced model is constrained to evolve in an affine subset of the full configuration space, for instance when a large finite element model is reduced by means of a linear combination of modes selected by proper orthogonal decomposition (POD). Such approaches are common in electrical engineering, molecular simulation, or mechanics [1], recently aimed at real-time simulation [2]. The term *nonlinear model reduction* generally refers to the reduced modeling of a nonlinear system, yet relying on a linear (affine) reduced description. Nevertheless, it is apparent that many models evolve on essentially nonlinear manifolds, in particular those involving large amplitude rotations. Rotations, prominent in many mechanical systems, are the paradigmatic example of a nonlinear motion inefficiently described by a linear geometric model. For instance, in three dimensions, nine Cartesian components are required to describe an orthogonal matrix, while only three parameters, e.g. Euler angles, suffice to parametrize a rotation.

In the field of statistical learning and pattern recognition, there has been a very intense research activity over the last decade in nonlinear dimensionality reduction (NLDR) of data sets [3]. These techniques extend the classical ideas of multi-dimensional scaling (MDS) or principal component analysis (PCA) to detect nonlinear hidden correlations in data lying in high dimensional spaces. These methods provide low-dimensional embeddings of the data, often shedding light on very complex phenomena. These methods have found applications in classification, automatic perception, climate science [4], the study of the conformation dynamics of molecules [5, 6], or galaxy spectra classification [7]. We have exploited elsewhere these methods for the meshfree thin shell analysis [8], and to analyze the stroke kinematics of a micro-organism [9].

Here, we explore the application of NLDR techniques to the reduced modeling of mechanical systems. In this first study, we restrict ourselves to conservative mechanical systems, and to the forward problem. We follow the classical method of snapshots, in which a training set of simulations, representative of the behavior of the system,

*marino.arroyo@upc.edu, +34 93 4011805

is used as the basis for the reduced model. Unlike linear approaches such as the POD (a direct application of PCA), here we consider a nonlinear reduced description, and consequently cannot parametrize the reduced model as a linear combination of modes. We proceed in three steps. First, the snapshots are viewed as a high-dimensional ensemble representative of the configurations of the system, the intrinsic dimensionality of the underlying slow manifold is detected, and the snapshots are embedded in low-dimensions by NLDR. Note that a mere embedding of the snapshots is not enough for our purpose, as it does not provide a parametrization of the slow manifold. For this reason, as a second step, we construct a smooth parametrization of the slow manifold (a map from the low-dimensional embedding to high dimensions) with local maximum-entropy approximants [10]. A number of key features of this approximation method are essential for the generality and success of the proposed method, namely its ability to provide smooth approximations from scattered unstructured data in any spacial dimension, adapted to the output of NLDR, and to filter noisy data robustly. As the third step, the embedding dimensions are then viewed as generalized coordinates of the system, in terms of which we express the dynamics. Although the real system evolves in a high dimensional space and departs from the slow manifold during its dynamics, we constrain the dynamics of the reduced system to the slow manifold.

There have been extensions beyond linear representations of reduced models, *à la* PCA, in computational mechanics. For instance, POD methods [11] address high-dimensional problems by finding iteratively new basis functions of separated variables, resulting in a nonlinear approximation method. Farhat and co-workers have proposed methods to interpolate parametrized reduced models, which form a nonlinear manifold [12].

The present work is organized as follows. Section 2 describes our model system, nonlinear elastodynamics, and succinctly its space discretization. Section 3 outlines the methodology proposed here. We formulate in Section 4 the Lagrangian mechanics of a system constrained to evolve in a manifold described by a set of generalized coordinates. We exercise the discrete Lagrangian mechanics formalism to derive variational time-integration schemes for systems with configuration-dependent mass. In Section 5, we develop a proof-of-concept example, highlighting the potential and limitations of reduced order modeling based on NLDR. Finally, we put forward some remarks and conclusions in Section 6.

2. Model system: nonlinear elastodynamics

We consider nonlinear elastodynamics as a model system to illustrate the proposed nonlinear model reduction method. Given the undeformed elastic body $\Omega_0 \subset \mathbb{R}^n$, where n is the spatial dimension of the problem, subject to body forces, with prescribed deformation on part of its boundary $\partial\Omega_0^D$, and prescribed tractions in the rest of the boundary of the body $\partial\Omega_0^N$, the goal is to obtain the deformation mapping $\varphi : \Omega_0 \longrightarrow \mathbb{R}^n$ satisfying balance of linear momentum and the boundary conditions:

$$\begin{aligned} \rho_0\, \ddot{\varphi} - \mathrm{Div}\, P - \rho_0\, b = 0 \qquad & \text{in } \Omega_0 \\ \varphi = \overline{\varphi} \qquad & \text{on } \partial\Omega_0^D \\ P\, N = \bar{t} \qquad & \text{on } \partial\Omega_0^N, \end{aligned}$$

with initial conditions $\varphi|_{t=0} = \bar{\varphi}_0$, and $\dot{\varphi}|_{t=0} = \dot{\bar{\varphi}}_0$ in Ω_0. Here Div denotes the divergence in material coordinates, P is the first Piola-Khirchhoff stress tensor, b the body force per unit mass, N the unit outward normal to the boundary of the undeformed body, $\bar{t}$ is the prescribed traction per unit undeformed area, ρ_0 is the mass density in the reference configuration, and $\overline{\varphi}$ is the prescribed deformation mapping in part of the boundary. These equations need to be supplemented by the constitutive relation, which in hyper-elastic materials takes the form

$$P(X) = \frac{\partial W}{\partial F}(D\varphi(X)), \quad \forall X \in \Omega_0.$$

Here $W(F)$ is the free energy density of the material, whose argument is the deformation gradient, i.e. the derivative of the deformation mapping $D\varphi$. For definiteness, we consider here a compressible neo-Hookean material with strain-energy density

$$W(F) = \frac{1}{2}\lambda \ln^2(J) + \frac{1}{2}\mu\left(\mathrm{tr}(F^T F) - n\right) - \mu \ln(J),$$

where $J = \det(F)$, and λ and μ are the Lamé coefficients.

The standard finite element discretization of this problem relies on the weak form of these equations and the finite element interpolation of the deformation mapping and the test functions. Alternatively, the discrete equations of motions can be derived following a Lagrangian formalism, by discretizing directly the kinetic and the potential energies. The total potential energy of the system is

$$V[\varphi] = \int_{\Omega_0} W(D\varphi)\, dV - \int_{\Omega_0} \rho_0\, b \cdot \varphi\, dV - \int_{\partial\Omega_0^N} \bar{t} \cdot \varphi\, dS,$$

whereas the kinetic energy is

$$T[\dot{\varphi}] = \frac{1}{2}\int_{\Omega_0} \rho_0\, \dot{\varphi}\cdot\dot{\varphi}\, dV.$$

The deformation mapping is discretized as

$$\varphi_h(X,t) = \sum_i \varphi_i(t)\, N_i(X), \tag{1}$$

where φ_i denote the nodal positions and $N_i(X)$ are the basis functions defined in the undeformed body. Consequently, the Lagrangian velocity field is

$$\dot{\varphi}_h(X,t) = \sum_i \dot{\varphi}_i(t)\, N_i(X). \tag{2}$$

We denote by $x = (\varphi_1, \varphi_2, \dots) \in \mathbb{R}^D$ the array containing all the nodal positions. Plugging Eqs. (1,2) into the potential and kinetic energy functionals, we obtain the discrete potential and kinetic energy functions $V_h(x) = V[\varphi_h]$ and $T_h(\dot{x}) = T[\dot{\varphi}_h]$, which also involve numerical quadrature. The out-of-balance forces at node b can be computed with the usual expression

$$\frac{\partial V_h}{\partial \varphi_b} = \int_{\Omega_0} P\,\nabla_0 N_b\, dV - \int_{\Omega_0} N_b\,\rho_0\, b\, dV - \int_{\partial\Omega_0^N} N_b\,\bar{t}\, dS,$$

where ∇_0 is the gradient with respect to the reference coordinates. The kinetic energy adopts the simple expression

$$T_h(\dot{x}) = \frac{1}{2}\dot{x}^T M \dot{x},$$

where the mass matrix components are

$$M_{aibj} = \delta_{ij}\int_{\Omega_0} \rho_0 N_a N_b\, dV.$$

3. Point-set manifold analysis

We concisely describe here the main steps required to describe the data-driven slow manifold $\mathcal{M}$, namely (1) the computation of a low-dimensional embedding of a training set of snapshots, and (2) the definition of a smooth parametrization of the slow manifold embedded in high dimensions. Before performing NLDR, the training set needs some preprocessing, described in Section 3.1, and the intrinsic dimension of the data d can be estimated by various techniques for a better-targeted reduction, Section 3.2. Dimensionality reduction is described in Section 3.3, and the parametrization of the slow manifold in Section 3.4.

3.1. Preparing the training ensemble

We assume now that, on the basis of an offline set of calculations, we have a configuration ensemble made out of simulation snapshots. In practice, a decimation step is necessary to define a workable training set, which we denote by $X = \{x_1, x_2, \dots, x_N\} \subset \mathbb{R}^D$. We define the ensemble such that any two snapshots are at least a cut-off distance away from each other, following a standard procedure in neural network analysis known as vector quantization [3], although many other more sophisticated techniques are available.

Unlike PCA, which requires the snapshots to be full vectors in $\mathbb{R}^D$ to avoid a lossy compression of the data, see Eq. (3) below, the method presented here allows us to prune the snapshots into coarser representations before we apply NLDR. For instance, it is easy to simplify systematically a large finite element mesh to a set of landmark points sampling the boundary, edges, or the skeleton, and thus decrease D significantly. This allows us to avoid the manipulation and storage of large amounts of data. For instance, with a 3D mesh with 10000 nodes and 10^5 snapshots, the unpruned training set requires about 22 GB of memory.

3.2. Intrinsic dimension

In PCA, the decay of the spectrum of the covariance matrix provides valuable information about the dimension of the affine hull of the data. Nonlinear dimensionality reduction methods require guessing the intrinsic dimension d of the underlying nonlinear manifold $\mathcal{M}$, a much more difficult task. The notion of intrinsic dimension is not clear-cut, and for data infected by noise, a generic situation, it depends on the scale of observation. It can be loosely defined as the minimum number of variables needed to describe the system behavior within a confidence level. The dimension can be estimated *a priori*, or the appropriateness of a guess can be assessed *a posteriori*.

A number of methods have been proposed to estimate d. We consider here the correlation dimension of the full data set X, which estimates the dimension by tracking the growth of the number of samples within a ball of increasing radius, and local PCA, in which the affine dimension is estimated for patches of X of a certain size located at different positions, as locally a manifold should look linear. The interested reader is referred to [3], and references therein. We discuss in Section 3.5 the *a posteriori* evaluation of the dimension.

3.3. Dimensionality reduction

We describe now how to embed the training set $X \subset \mathbb{R}^D$ in dimension d, i.e. find a reduced representation of the snapshots $Q = \{q_1, q_2, \ldots, q_N\} \subset \mathbb{R}^d$. We consider a linear method, PCA [13], and two NLDR methods, Isomap [14] and an iterative method that we call Nonlinear Locally Isometric Embedding (NLIE), closely related to others in the field (see Appendix C).

Linear Model Reduction. PCA is a classical linear dimensionality reduction method, widely used in model order reduction, e.g. by [1] in the context of nonlinear solid dynamics. First, the singular value decomposition of the covariance matrix, a $D \times D$ matrix, is computed. Then, a matrix with the eigenvectors associated to the d largest eigenvalues, $P \in \mathbb{R}^{D\times d}$, is defined. The low dimensional embedding is computed through the projection $q_a = P^T(x_a - \bar{x})$, where $\bar{x} = (1/N)\sum_{a=1}^{N} x_a$. Note that this projection can be applied to out-of-sample high dimensional points, i.e. points not belonging to the training set X. Conversely, PCA provides a direct parametrization of the slow manifold (a hyperplane) as

$$\mathbb{X}(q) = Pq + \bar{x}, \quad \forall q \in \mathbb{R}^d. \tag{3}$$

Figure 1 illustrates how PCA approximates the data with the $d-$dimensional affine manifold that best describes the variance. However, if the data lies on a curved manifold, PCA provides in general a poor description unless a dimension larger than the intrinsic dimension is considered.

Multi-dimensional scaling (MDS) provides a complementary point of view to PCA. The low dimensional embedding is found by minimizing the discrepancy between the distance matrix between the points in low dimensions and the distance matrix in high-dimensional space. If the distance metric is Euclidean, this method coincides with PCA. However, it involves the singular value decomposition of a $N \times N$ full matrix, which can be beneficial or not depending on the application.

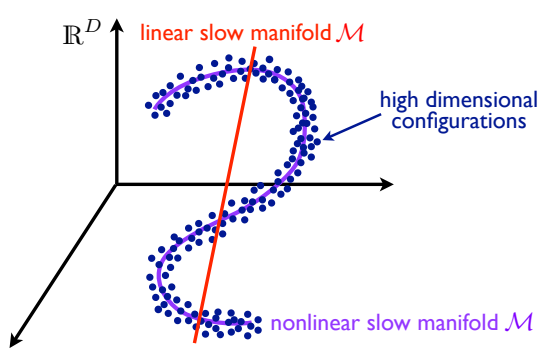


Figure 1: Sketch of the the slow manifold obtained with linear and nonlinear dimensionality reduction methods from a data set with low-dimensional nonlinear correlations (dark blue points). A linear method (red line) only captures the variability if the number of dimensions is larger than the intrinsic dimension, here $d = 1$, whereas the nonlinear method, at the expense of extra complexity, can describe the system optimally in terms of dimension. Note that NLDR algorithms do not provide a description of the nonlinear slow manifold (purple line), but only a low-dimensional embedding, see Fig. 2.

Nonlinear Model Reduction. Nonlinear dimensionality methods try to identify automatically nonlinear, low-dimensional correlations in the high-dimensional data. One of the most popular methods, Isomap, tries to isometrically embed the high-dimensional data in $\mathbb{R}^d$ through MDS. The key point is that the distance matrix of the points in X is a graph-based approximation to the geodesic distance, not the Euclidean distance. Isomap has

been shown to be robust for data polluted with noise, or for non-uniformly distributed data points. However, even for surfaces in three dimensions, we know as a corollary of Gauss's Theorema Egregium [15] that it is not possible to isometrically embed in $\mathbb{R}^2$ a surface with non-zero Gaussian curvature. This fact leads to a frustration in the algorithm when handling highly curved manifolds, which can lead to instabilities and generate collapsed embeddings. As we shown in Section 5, this spurious behavior manifests itself in practical applications, and demands more robust methods. It can be dealt with by partitioning the training set into geometrically simpler pieces [8]. Partitioning is also necessary when dealing with manifolds of complex topology. Alternatively, we combine Isomap, whose output is taken as an initial seed, with an iterative in-house algorithm, NLIE. In NLIE, the embedding is obtained by minimizing of a cost stress energy function, which measures the difference between the distances computed in the embedding and in the ambient space. The metric used to measure distances in the high-dimensional space is an approximation to the geodesic distance (the shortest path distance on a graph built with $k-$nearest neighbors), as in Isomap. The main difference lies in the definition of the stress energy, which is computed taking into account only a moderate number of neighbors to retain accuracy and sparsity (see Appendix C).

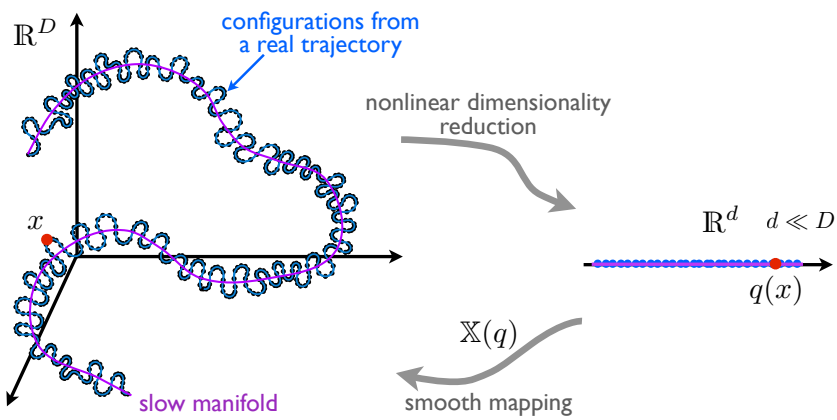


Figure 2: Sketch of the method to smoothly parametrize point-set manifolds. The set of high-dimensional snapshots X (dark blue points) are embedded in low dimensions (light-blue points) by NLDR methods, possibly applied to the pruned snapshots for computational convenience. With the embedding at hand, the slow manifold hidden in the training set X is represented with a smooth, data-driven parametrization $\mathbb{X} : \mathscr{A} \subset \mathbb{R}^d \longrightarrow \mathbb{R}^D$. It is also possible to find the low-dimensional representation $q(x) \in \mathbb{R}^d$ of a new out-of-sample configuration $x \in \mathbb{R}^D$, where $q(x)$ is the pre-image by $\mathbb{X}$ of the closest-point projection of x on $\mathcal{M}$, found solving a nonlinear least-squares problem [8].

3.4. Parametrization of the slow nonlinear manifold

In reduced order modeling, we expect the system to abide by the reduced manifold, although the trajectories of the full system do not need to lie on it. This simplifying assumption introduces some error, compensated by a much faster simulation time. Figure 2 illustrates this idea, and the general pipeline of the current method. It is clear that the low-dimensional space where the training set is embedded serves as a convenient parametric space. The embedded points in Q are in general unstructured, and, although we show here examples in $d = 2$, the methodology is applicable to higher-dimensional embedded manifolds. Thus, a general method to compute on point-set manifolds demands a smooth approximation scheme for general unstructured nodes in multiple dimensions. The need for smoothness in the present application is shown in the Section below, e.g. Eq. (12). Here, we consider the local *max-ent* approximants [10], a general meshfree method satisfying these requirements.

Let $p_a(q)$ denote the local *max-ent* basis functions associated to the node set Q, defined in a subset of the convex hull of Q, $\mathscr{A} \subset \mathbb{R}^d$. We parametrize the manifold as

$$\begin{aligned} \mathbb{X} : \mathscr{A} &\longrightarrow \mathbb{R}^D \\ q &\longmapsto \sum_a p_a(q)\, x_a, \end{aligned} \tag{4}$$

where $x_a \in \mathbb{R}^D$ are the full-sized snapshots, irrespective of the fact that Q may have been found on the basis of the pruned snapshots.

3.5. Reconstruction error

With a parametrization of the slow manifold at hand, it is possible to evaluate its quality by comparing high dimensional data with their reconstruction through the parametrization. Ideally, this is done with a test set $Z = \{z_1, \dots, z_M\} \subset \mathbb{R}^D$, different and possibly larger than the training set, sampling $\mathcal{M}$. We can then define the relative reconstruction error

$$e = \frac{1}{M} \sum_{a=1}^{M} \frac{|z_a - \mathbb{X}(q(z_a))|}{|z_a|}, \tag{5}$$

where $q(x)$ maps high-dimensional points to low-dimensional representations, see Fig. 2 and [8]. For PCA, we have

$$e = \frac{1}{M} \sum_{a=1}^{M} \frac{\left|z_a - PP^T(z_a - \bar{x}) - \bar{x}\right|}{|z_a|}. \tag{6}$$

4. Lagrangian mechanics in generalized coordinates

4.1. Constrained dynamics

Let $x \in \mathbb{R}^D$ denote the position in Cartesian coordinates of a mechanical system described by its Lagrangian as follows

$$L(x, \dot{x}) = T(\dot{x}) - V(x) = \frac{1}{2} \dot{x}^T M \dot{x} - V(x),$$

where T is the kinetic energy, V the potential energy and M is the constant mass matrix. The corresponding Euler-Lagrange equations can be written as

$$M\ddot{x} = -\frac{\partial V}{\partial x}(x).$$

We are interested in the constrained dynamics on a configuration manifold $\mathcal{M} \subset \mathbb{R}^D$ of dimension $d \ll D$. Let $q \in \mathscr{A} \subset \mathbb{R}^d$ be the generalized coordinates given by NLDR as described above. We parametrize $\mathcal{M}$ in terms of these generalized coordinates as $x = \mathbb{X}(q)$. By the chain rule, $\dot{x} = D\mathbb{X}(q)\, \dot{q}$, and therefore we can formulate the constrained dynamics as

$$L(q, \dot{q}) = \frac{1}{2} \dot{q}^T \widetilde{M}(q)\, \dot{q} - \widetilde{V}(q),$$

where

$$\widetilde{M}(q) = D\mathbb{X}(q)^T M\, D\mathbb{X}(q) \quad \text{and} \quad \widetilde{V} = V \circ \mathbb{X}. \tag{7}$$

Since the mass matrix is in general configuration-dependent, this is an instance of non-separable Lagrangian. By Hamilton's principle, the Euler-Lagrange equations of the reduced system, expressing the balance of linear momentum in curvilinear coordinates, are

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = 0.$$

In the special case of PCA (or other linear methods), the manifold parametrization is affine, $D\mathbb{X} = P$, and consequently the reduced mass matrix is not position dependent, $\widetilde{M} = P^T M P$.

4.2. Discrete Lagrangian mechanics

We resort to variational principles to integrate in time the Euler–Lagrange equations; full details can be found in [16, 17] and references therein. Starting with the action functional

$$S[q] = \int_{t_0}^{t_1} L(q, \dot{q})\, dt,$$

for the system's trajectory $q(t)$ in $\mathscr{A}$ satisfying the initial conditions, variational integrators are based on Hamilton's principle, which states that the trajectories are stationary points of the action, $\delta S[q, \delta q] = 0$ for all admissible δq.

In time-discrete Lagrangian mechanics (one step methods), the action integral S is replaced by an action sum

$$S_d = \sum_{j=0}^{n-1} L_d(q_j, q_{j+1}),$$

where $L_d : \mathcal{M} \times \mathcal{M} \to \mathbb{R}$ is the discrete Lagrangian, a function of two points $q_0 = q(t_0)$, $q_1 = q(t_1) \in \mathscr{A}$ approximating the action in an interval as

$$L_d(q_0, q_1) \approx \int_{t_0}^{t_1} L(q, \dot{q})\, dt. \tag{8}$$

The discrete Euler-Lagrange equations –the time-integration algorithm–, follow from making the action sum stationary, e.g. $\partial S_d / \partial q_k = 0$, and are given by

$$D_1 L_d(q_k, q_{k+1}) + D_2 L_d(q_{k-1}, q_k) = 0. \tag{9}$$

The accuracy of the variational integrator is given by the order of the quadrature in Eq. (8). Specific numerical integration schemes can be obtained from this formalism. For concreteness, we derive here the discrete Euler-Lagrange equations for the generalized midpoint rule with configuration-dependent mass. Additionally, in Appendix A, the resulting expressions for the trapezoidal rule for a configuration-dependent mass are given. For the generalized midpoint rule, the discrete Lagrangian approximating the action integral between q_0 and q_1 is given by

$$\begin{aligned} L_d(q_0, q_1) &= \Delta t\, L\Big((1-\alpha)q_0 + \alpha q_1, \frac{q_1 - q_0}{\Delta t}\Big) \\ &= \Delta t \left\{ \frac{1}{2} \left(\frac{q_1 - q_0}{\Delta t}\right)^T \widetilde{M}\Big((1-\alpha)q_0 + \alpha q_1\Big)\Big(\frac{q_1 - q_0}{\Delta t}\Big) - \widetilde{V}\Big((1-\alpha)q_0 + \alpha q_1\Big)\right\}, \end{aligned} \tag{10}$$

where $\alpha \in [0, 1]$. From Eqs. (9,10), we obtain the following equation for q_{k+1}

$$\begin{aligned} 0 = \frac{\partial S_d}{\partial q_k} \;=\; & \Delta t \left\{ (1-\alpha) D_1 L\Big((1-\alpha)q_k + \alpha q_{k+1}, \frac{q_{k+1} - q_k}{\Delta t}\Big) - \frac{1}{\Delta t} D_2 L\Big((1-\alpha)q_k + \alpha q_{k+1}, \frac{q_{k+1} - q_k}{\Delta t}\Big)\right\} \\ & + \Delta t \left\{ \alpha D_1 L\Big((1-\alpha)q_{k-1} + \alpha q_k, \frac{q_k - q_{k-1}}{\Delta t}\Big) + \frac{1}{\Delta t} D_2 L\Big((1-\alpha)q_{k-1} + \alpha q_k, \frac{q_k - q_{k-1}}{\Delta t}\Big)\right\} \\ = \; & \Delta t\,(1-\alpha) \left\{ \frac{1}{2} \left(\frac{q_{k+1} - q_k}{\Delta t}\right)^T D\widetilde{M}\Big((1-\alpha)q_k + \alpha q_{k+1}\Big)\Big(\frac{q_{k+1} - q_k}{\Delta t}\Big) - D\widetilde{V}\Big((1-\alpha)q_k + \alpha q_{k+1}\Big)\right\} \\ & + \Delta t\, \alpha \left\{ \frac{1}{2} \left(\frac{q_k - q_{k-1}}{\Delta t}\right)^T D\widetilde{M}\Big((1-\alpha)q_{k-1} + \alpha q_k\Big)\Big(\frac{q_k - q_{k-1}}{\Delta t}\Big) - D\widetilde{V}\Big((1-\alpha)q_{k-1} + \alpha q_k\Big)\right\} \\ & - \widetilde{M}\Big((1-\alpha)q_k + \alpha q_{k+1}\Big)\Big(\frac{q_{k+1} - q_k}{\Delta t}\Big) + \widetilde{M}\Big((1-\alpha)q_{k-1} + \alpha q_k\Big)\Big(\frac{q_k - q_{k-1}}{\Delta t}\Big). \end{aligned} \tag{11}$$

These calculations require the directional derivative of the reduced mass matrix, and derivative of the reduced potential energy, computed as follows from the product and the chain rule

$$D\widetilde{M}(q)\,\dot{q} = \mathbb{Y}^T(q, \dot{q})\, M\, D\mathbb{X}(q) + D\mathbb{X}(q)^T M\, \mathbb{Y}(q, \dot{q}),$$

with

$$\mathbb{Y}(q, \dot{q}) = D^2\mathbb{X}(q)\,\dot{q}, \tag{12}$$

and

$$D\widetilde{V}(q) = DV(\mathbb{X}(q))\, D\mathbb{X}(q).$$

It is also possible to obtain an expression for the generalized velocities associated to the Cartesian velocities at a point in $\mathcal{M}$ and tangent to it

$$\dot{q} = \Big[D\mathbb{X}(q)^T\, D\mathbb{X}(q)\Big]^{-1} D\mathbb{X}(q)^T\, \dot{x}. \tag{13}$$

It is clear from Eq. (12) that this formulation requires a smooth representation of the slow manifold.

From a practical viewpoint, the choice $\alpha = 0$ is most convenient since then Eq. (11) is merely quadratic in q_{k+1}. Otherwise, the resulting algorithm is implicit in q_{k+1} in a very nonlinear way, requiring third order derivatives of $\mathbb{X}(q)$ for an iterative solution with Newton's method. The implementation details for $\alpha = 0$ in combination with Newton's method are given in Appendix B.

5. Proof-of-concept example: a neo-Hookean oscillator

We show here the capabilities of the present method in a proof-of-concept problem, the dynamics of a hyperelastic neo-Hookean body undergoing large deformations. We compare the proposed nonlinearly reduced dynamics against the standard reduced dynamics from a linear method (PCA). The geometry of the object and mesh are shown in Fig. 3.

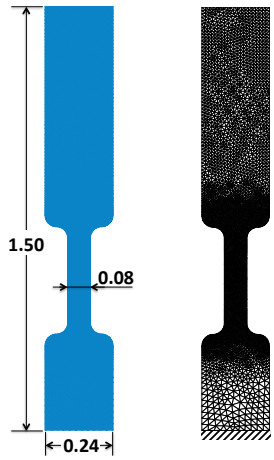


Figure 3: Sketch for the neo-Hookean hyperelastic solid in 2D (left) and the mesh used to compute the elastodynamics (right). The object is clamped at the bottom, and subject to gravity. The material parameters are $\lambda = 1000$ and $\mu = 500$, and the model is discretized by 9428 linear triangular elements and 4934 nodes.

5.1. Training set

We simulated 35 trajectories for different initial conditions (position and velocity), with an explicit mid-point rule integration scheme. For each trajectory, a total number of $4 \cdot 10^6$ time steps were computed and sampled with a frequency of 1/1000, resulting in 4000 snapshots per trajectory. In an offline process, the first 500 snapshots were deleted to avoid poorly sampled regions arising from unlikely starting points. Further, due to the symmetry of the problem, each retained snapshot was duplicated and mirrored. As a result, a total of 245,000 full configurations in $D = 4934 \times 2$ were obtained. These snapshots were then pruned by keeping only the nodes at the boundary, resulting in a simplified snapshots in dimension $D = 868$, approximately one tenth of D. In pruning the snapshots, one needs to balance compression and fidelity. For instance, we found that keeping only 100 points along the skeleton resulted in a poorer description of the dynamics. Finally, we decimated the pruned training set, down to 50,000 configurations in $D = 868$.

5.2. Intrinsic dimension

As discussed in Section 3.5, we can *a priori* estimate the intrinsic dimension underlying the data set. Figure 4 illustrates how the correlation dimension (A) and the cumulative variance error of local PCA (B) provide hints, but not conclusive information for realistic training sets. These method are blind to the origin of the data. The intrinsic dimension appears to be between 2 and 4, depending on the scale of observation. We evaluate the dimension *a posteriori* (after dimensionality reduction on the parametrization of the slow manifold) by tracking the reconstruction error as a function of the dimension of the embedding, see Fig. 4C. With PCA, it is straightforward to increase the dimension, while for the nonlinear methods, whose objective is to keep d as small as possible, we have only constructed one- and two-dimensional slow manifolds. As expected for data with nonlinear correlations, it can be observed that the reconstruction error stemming from NLDR are significantly smaller than that given by PCA for a given dimension. The dimensionality estimations are consistent with the *a priori* methods. From our experience, the most practical method of dimensionality estimation is tracking the cumulative variance error of local PCA. It is fast, easily made parallel, can be performed on pruned data, and does not need an embedding of the training set.

5.3. Dimensionality reduction

Figure 5 shows the two- and three-dimensional embeddings of the training set by PCA, Isomap and NLIE. It is obvious from the three-dimensional embeddings that the data has highly nonlinear correlations, i.e. the slow manifold is highly curved. As a result, it is not surprising that Isomap collapses some parts of the training set. In contrast, NLIE gently unrolls the curved manifold by introducing local distortions that minimize the global stress energy. A close visual inspection of the three-dimensional embedding provided by PCA or NLIE suggests that the data lies on a "thick" elongated hyper-surface.

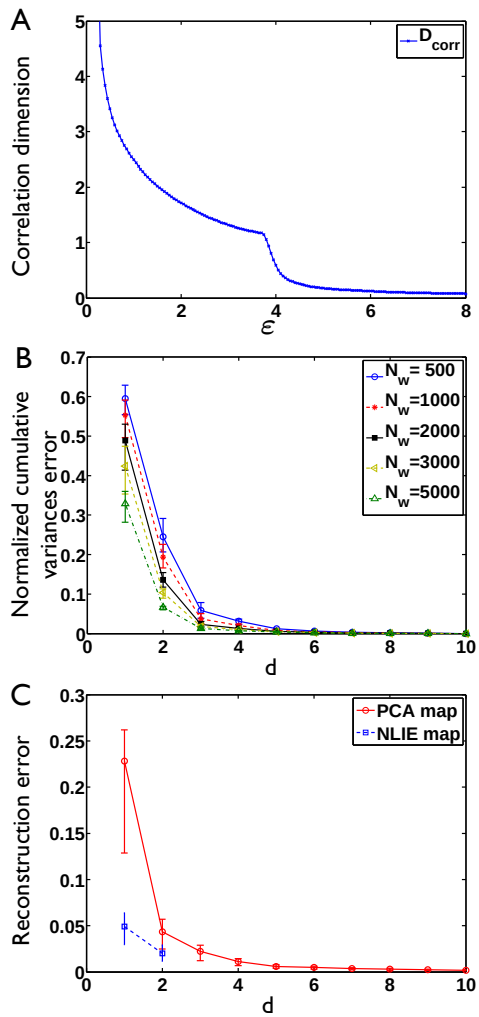


Figure 4: (A) The correlation dimension method considers a closed ball of radius ε at the center of each data point, and counts the number of points inside of this ball. The dimension is estimated by noting that the average number of counts $C(\varepsilon)$ should grow linearly with ε for 1D objects, quadratically for 2D entities, and so on, i.e. $C(\varepsilon) \propto \varepsilon^d$. The correlation dimension is then defined as $d_{corr}(\varepsilon) = d \log C(\varepsilon)/d \log \varepsilon$. It is a scale-dependent quantity that should be interpreted carefully a: at very large ε, $d_{corr} \approx 0$ as the data set looks like a single point, while for small ε, the dimension is overestimated by the presence of noise. (B) Mean of the normalized cumulative variance error from local PCA ($\pm$ 25th and 75th percentiles), $\left(1 - \sum_{i=1}^{d} \lambda_i\right) / \sum_{i=1}^{D} \lambda_i$, computed in small local subsets of the data where N_w is the window size. λ_i denote the eigenvalues of the local covariance matrix arranged in decreasing order. The intrinsic dimension d for each of these patches is selected such that it preserves a given fraction of the variance of the original data. In contrast with the global correlation dimension, this method provides a local estimation of the intrinsic dimension. (C) Mean of the reconstruction error as defined in Eqs. (5,6) ($\pm$ 25th and 75th percentiles). The reconstruction error is a measure of the dissimilarity between the original high-dimensional data points and their reconstruction through the slow manifold parametrization.

The one-dimensional embedding of PCA can be found by projecting along q_1 the two-dimensional embedding. It is clear from the figure that this method will collapse far-away conformations, destroying the geometric structure of the slow manifold. In contrast, the one-dimensional NLIE embedding provides a coarse, yet meaningful representation of the model.

We finally note that, in principle, it is possible to define a nonlinear parametrization of the slow manifold such as that in Eq. (4) on the basis of PCA embeddings. However, from our experience in this and other applications, these embeddings introduce larger distortions and are not robust for significantly curved training sets.

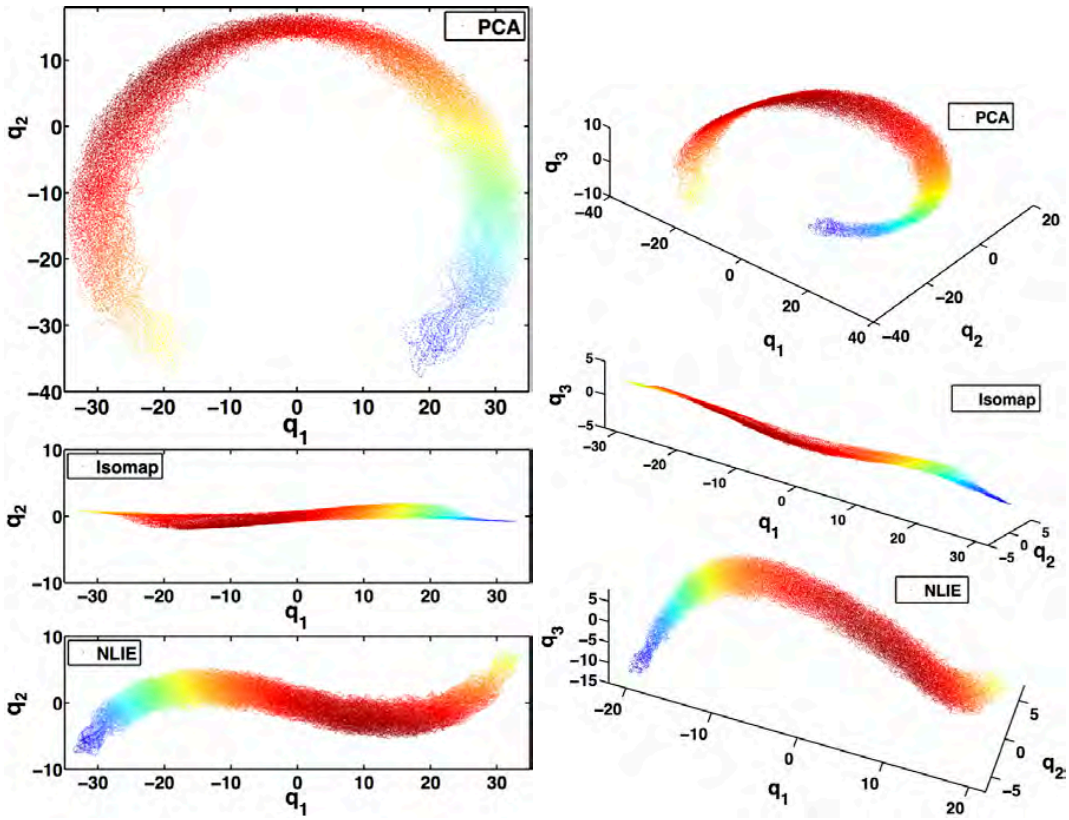


Figure 5: Two- and three-dimensional embeddings obtained with PCA (top), Isomap (center) and NLIE (bottom). The colormap, indicating the Euclidean distance to the first configuration of the training set, is only provided as visual guide.

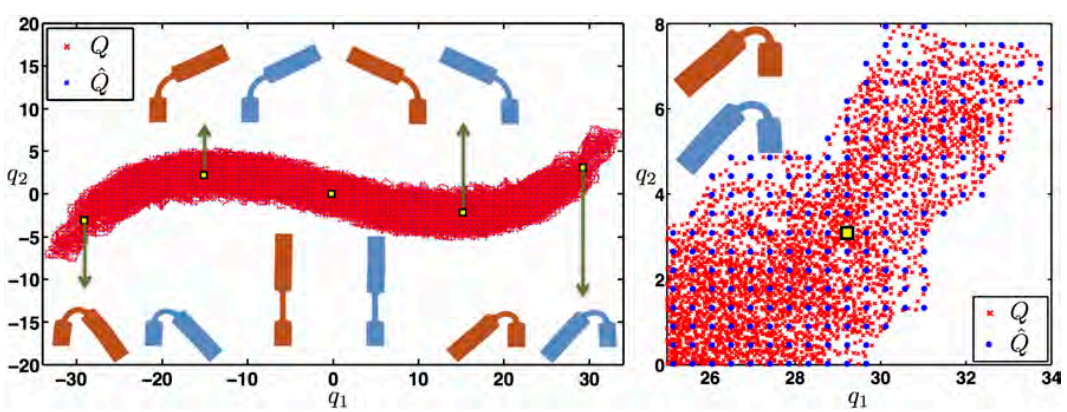


Figure 6: Two-dimensional embedding Q obtained from the original training set X with NLIE (red points), and subsampled node set $\hat{Q}$ (blue points), over the entire embedding (left). For some selected $\hat{q}_b$, we depict the corresponding configuration $\hat{x}_q$ obtained from the least squares fit, and the closest conformation from the original training set. A magnification of the upper-right corner of the embedding (right).

5.4. Parametrization of the slow manifold

We follow Eq. (4) to parametrize smoothly the slow manifold from the embedding, $\mathbb{X}(q)$, with LME basis functions. Details about the calculation of these meshfree approximants and their spatial derivatives can be found in [10, 18]. The noise filtering can be controlled with the width of the LME basis functions. Figure 6 shows in more detail the two-dimensional embedding found with NLIE, where the red points represent the set Q. An obvious drawback of defining $\mathbb{X}(q) = \sum_{a=1}^{N} p_a(q) x_a$ is that N can be very large, here 50,000, and therefore the

evaluation of this mapping, required at each time-step, can be quite expensive. In fact, this mapping parametrizes a relatively simple slow manifold, and the large number of samples is only required to provide a statistically representative embedding. For computational convenience, we subsample the description of the slow manifold as follows. We divide the embedding space in equally spaced bins of size $h_{bin} = L/n_{bin}$, where L is the dimension of the embedding along q_1 and n_{bin} the number of bins in this direction. h_{bin} should be commensurate to the geometrical features of the slow manifold. This defines a regular grid, from which we retain the points lying in well-sampled regions of the embedding, the blue points in Fig. 6, that we denote by $\hat{Q} = \{\hat{q}_1, \hat{q}_2, \ldots, \hat{q}_{\hat{N}}\}$ with $\hat{N} \ll N$. We then define the local maximum-entropy basis functions associated to $\hat{Q}$, $\hat{p}_b(q)$, with the locality parameter $\beta = 1/h_{bin}^2$ [10], and fit the slow manifold with a least-squares criterion (see also [8])

$$\min_{\{\hat{x}_1,\ldots\hat{x}_{\hat{N}}\}} \sum_{a=1}^{N} \left\| x_a - \sum_{b=1}^{\hat{N}} \hat{p}_b(q_a)\hat{x}_b \right\|^2 .$$

In the example shown in Fig. 6, we selected $n_{bin} = 150$, leading to a subsampling of the slow manifold from $N = 50000$ to $\hat{N} = 2115$ control points.

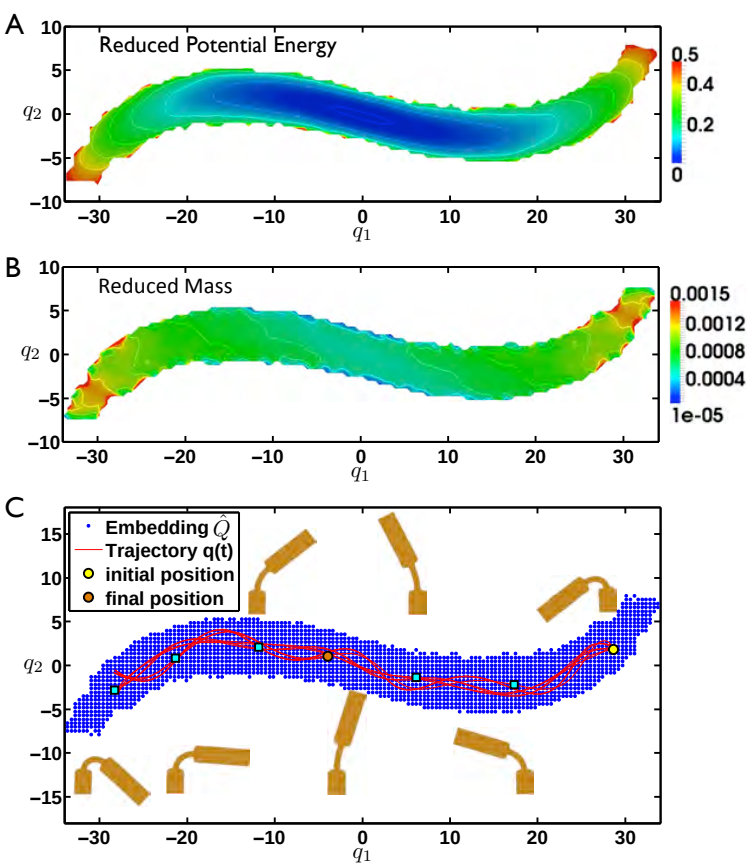


Figure 7: (A) Reduced potential energy $\widetilde{V}(q)$. (B) Total reduced mass $\sum_{i,j} \widetilde{M}_{ij}(q)$. (C) Two-dimensional reduced trajectory in the embeding from NLIE, over seven large amplitude oscillations of the system. The initial, final, and selected snapshot positions are highlighted.

We can time-integrate the reduced dynamics with the methods outlined in Section 4.2 and the smooth mapping $\mathbb{X}(q) = \sum_{b=1}^{\hat{N}} \hat{p}_b(q)\hat{x}_b$, leading to a numerical trajectory in the low-dimensional space $q(t)$. The nonlinearly reduced dynamics are fully determined by the effective potential energy $\widetilde{V}(q)$ and the reduced mass $\widetilde{M}(q)$. Figure 7A shows the effective potential, whereas Fig. 7B shows the total reduced mass in the embedding. It can be observed that the reduced mass, which is in fact the square mass-weighted L_2 norm of the derivative of the slow manifold parametrization, see Eq. (7), is nearly constant except at the boundaries of the region of interest. At the boundary, it is not surprising that the quality of the mapping $\mathbb{X}(q)$ is degraded. Yet, these regions are visited seldom due to their high potential energy.

Figure 7C shows an example of reduced dynamics $q(t)$. Visually, these dynamics retain the qualitative features of the full, high-dimensional dynamics, with high amplitude oscillations. In the examples below, we consider both PCA- and NLIE-reduced dynamics. We always use a midpoint rule with $\alpha = 0$, with a time step $\Delta t = 10^{-4}$ in the reduced simulations (5 times larger than that of the full dynamics). For PCA, details about the time-integration are given in Appendix A, whereas for NLIE, due to the mass position-dependency, the algorithm is nonlinear and quasi-explicit (only the gradient of $V(x)$ is required). We use Newton's method, see Appendix B.

For a more quantitative assessment of the quality of the reduced dynamics, we compare them to the full dynamics, $x(t)$, with the following metric

$$\int_{t_0}^{t_1} D(x(t), \mathbb{X}(q(t)))\, dt = \int_{t_0}^{t_1} \left(\int_{\Omega_0} \left\| \varphi_h(x(t)) - \varphi_h(\mathbb{X}(q(t))) \right\|^2 dV \right)^{1/2} dt,$$

where here $\varphi_h(x)$ denotes the finite element deformation mapping defined from the nodal positions x. In practice, we sample the time-integral at the time-steps $t_j = j\Delta t$, and compute a non-dimensional distance between trajectories $\mathcal{D}$ as

$$\mathcal{D} = \frac{\sum_{j=0}^{n} \sqrt{(x_j - \mathbb{X}(q_j))^T M\, (x_j - \mathbb{X}(q_j))}}{\sum_{j=0}^{n} \sqrt{x_j^T M\, x_j}},$$

where M is the mass matrix of the elastic body computed at the reference configuration.

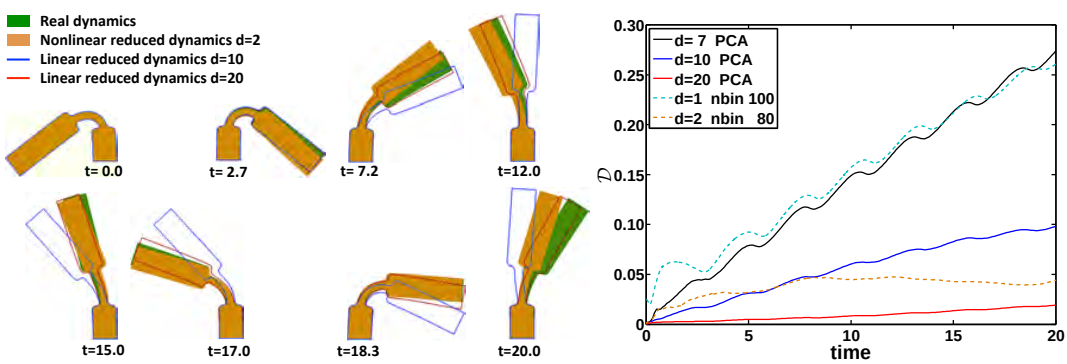


Figure 8: Snapshots during the time evolution of a high-dimensional trajectory $x(t)$, and the reduced trajectories $\mathbb{X}(q(t))$ obtained from PCA and NLIE (left). The quality of the reduced dynamics is measured by the relative cumulative distance between $x(t)$ and $\mathbb{X}(q(t))$ (right).

Figure 8 compares, both visually with conformations at selected instants, and quantitatively with $\mathcal{D}$, the full dynamics with the reduced dynamics obtained either by PCA with various dimensions, or by NLIE with dimensions one and two. Opposite to what could be expected from the intrinsic dimensionality analysis, the results indicate that more than 10 PCA modes are needed to capture the dynamics of the system. With only one nonlinear generalized coordinate, we obtain a poor description of the system, comparable to 7 PCA modes. This is not surprising, since the large amplitude oscillations lead to a highly nonlinear configuration manifold, and therefore PCA needs to properly describe its affine hull for reasonable results. As the numbed of PCA modes increases, we obtain accurate dynamics. Strikingly, with only two nonlinear dimensions, we obtain results that lie between those given by 10 and 20 PCA modes. The results suggest that in this example, the affine hull of the slow manifold has tens of dimensions, highlighting the benefit of nonlinear reduced descriptions.

Remark. A major conclusion thrown by the numerical experiments is that the reconstruction error is not a trustable measure of the reduced dynamics quality. This is because of the reconstruction error only notices for statistically meaningful differences between a given configuration x and its representation $\mathbb{X}(q)$, it is statistical based instead of physical based. In a nonlinear material it is well known that there is a huge transfer of energy from the slow modes to the fastest ones, then small variations in the body shape are related with important energy contributions to the dynamics behavior.

5.5. Energy leak of the slow modes

Figure 9 shows the energy evolution of the full (left) and the reduced (right) dynamics over seven oscillations. It can be observed that, when the system visits its symmetric configuration, the potential energy lies close to its minimum while the kinetic energy is highest. Around this point, high frequency oscillations in the energy can

be observed, suggesting a transfer to high frequency modes consistent with the visualization of the dynamics. The reduced dynamics exhibit similar features, but with only one transversal dimension to the main nonlinear coordinate, the features are significantly blunter. This observation raises the question of the long-time energy behavior of the reduced dynamics.

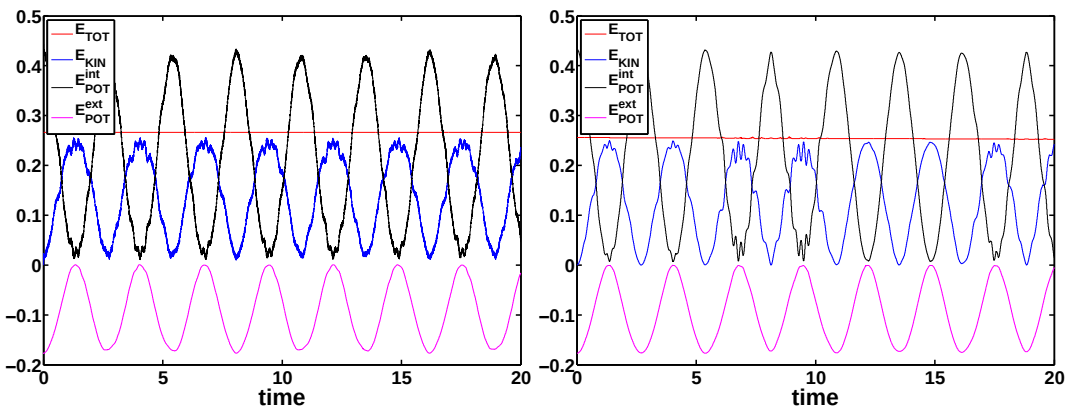


Figure 9: Energy evolution for the full system trajectory (left), and for the reduced trajectory with a 2-dimensional nonlinear embedding (right).

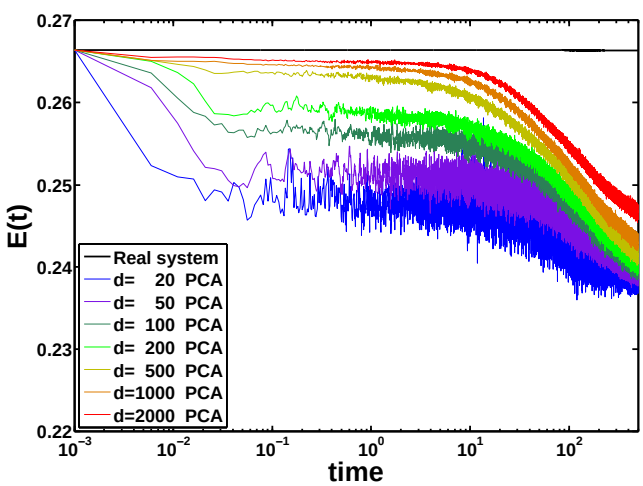


Figure 10: Total energy evolution of the full system and total energy projected onto the reduced models by PCA for different numbers of eigenvectors, $d = 20, 50, 100, 200, 500, 1000, 2000$.

To investigate numerically this issue, we consider a very long trajectory of the full system, and project the dynamics onto the PCA or nonlinearly reduced models. The projection onto the PCA modes is trivial. For the nonlinear model, we use the mapping $q(x)$ described in Fig. 2, together with Eq. (13) to compute $\dot{q}(x, \dot{x})$. We can then compute the full total energy,

$$E(t) = H(x(t), \dot{x}(t)) = V(x(t)) + T(\dot{x}(t)),$$

which is constant up to numerical accuracy since the system is conservative, and the reduced Hamiltonian

$$\widetilde{E}(t) = \widetilde{H}(q(x(t)), \dot{q}(x(t), \dot{x}(t))) = \widetilde{V}(q(x(t))) + \widetilde{T}(q(x(t)), \dot{q}(x(t), \dot{x}(t))).$$

Figure 10 shows the time-evolution of $E(t)$ and $\widetilde{E}(t)$ for $d = 20, 50, 100, 200, 500, 1000, 2000$ PCA modes over a very long simulation. Interestingly, there is a significant and monotonic flow of energy from the low modes into the high-frequency modes. Increasing the reduced dimension, effective at short times, only slightly reduces the

energy leak at long times. Once this energy is "thermalized" in the high-frequencies, it does seem to return to the low-frequency modes. Thus, the system described by the reduced coordinates is no longer conservative. These results suggest the need for modeling the net effect of the disregarded degrees of freedom on the reduced dynamics, as is done in the large eddy simulation of turbulence. The long experience in coarse-graining of molecular systems and non-equilibrium statistical mechanics could be useful in this regard [19].

6. Concluding remarks

We have proposed a novel method for reduced order modeling and exercised it on large-deformation nonlinear elastodynamics. In contrast with most approaches, the reduced model is not an affine one, as given e.g. by PCA or POD. Instead, we use modern NLDR methods to embed an ensemble of snapshots in low-dimensions, and then build from this embedding a smooth representation of the system in terms of nonlinear, data-driven, generalized coordinates. The underlying hypothesis is that many systems evolve on a nonlinear, low-dimensional slow manifold, and that by acknowledging the nonlinearity, it is possible to buid more efficient and insightful reduced models. We discuss in detail the practical aspects of the methodology, which requires extra data structures and procedures, but results in a much more compact description, which presumably leads to more insight. For instance, the reduced dynamics of the finite element equations of elastodynamics have a position-dependent mass, and we derive the corresponding variational time-integrators.

We have tested the method in a proof-of-concept example, a structure made out of a Neo-Hookean material undergoing large amplitude oscillations. By comparing the nonlinearly reduced dynamics with those generated by a linear methods, here PCA, our results suggest that nonlinearity reduces the number of dimensions required for an accurate description: two nonlinear generalized coordinates are equivalent to between 10 and 20 PCA modes. Below 10 PCA modes, the linear reduced model cannot capture even qualitatively the dynamics. Note that the PCA model is linear with regards to the structure of the reduced manifold, but fully nonlinear in that the underlying model is nonlinear elasticity. Our analysis of the intrinsic dimension of the ensemble of snapshots suggests that the dynamics lie on a slow manifold of dimension between 2 and 4. PCA only works if it adequately describes the affine hull of the slow manifold, which is of significantly higher dimension. In contrast, two nonlinear coordinates capture the essential features of the system. We have also shown that the nonlinearly reduced description can be coarsened in many ways to speed-up the evaluation of the slow manifold. Having a fast and optimally reduced description of the system is very attractive in some applications such as optimal control.

One of the limitations of the nonlinear reduction method is that increasing the dimension requires significantly larger training sets, so that the low-dimensional embedding of the snapshots is not too sparse. Otherwise, the parametrization of the slow manifold becomes problematic. In our proof-of-concept example, we attempted to reach $d = 3$, but found that the mapping $\mathbb{X}(q)$ was of poor quality (very large derivatives) close to the boundary of the region of interest, leading to instabilities in the time-integration. This issue can be resolved hierarchically. For instance, the two-dimensional embedding can be used to generate new training trajectories from poorly-sampled regions on the boundary, thereby improving the sampling of the slow manifold. According to our preliminary calculations, a subsequent dimensionality reduction process leads to better embeddings, also in higher dimensions. Our experience also suggests that, beyond mere geometric similarity, which cannot distinguish high-frequency but possibly highly energetic features, the dimensionality reduction could take into account information about the mechanics of the snapshots, such as their potential energy as an extra coordinate in x_a. We feel this may result in embeddings better suited for mechanical simulations.

Finally, we have shown that our Hamiltonian system of interest, when described in terms of a reduced model, is no longer conservative. We find that the energy leak is significant for long times, and cannot be effectively controlled by increasing the PCA dimension to 1000's. This suggests that reduced order simulation should be accompanied by modeling the net effect of the disregarded degrees of freedom on the reduced system.

Acknowledgments

We acknowledge the support of the European Research Council under the European Community's 7th Framework Programme (FP7/2007-2013)/ERC grant agreement nr 240487. MA acknowledges the support received through the prize "ICREA Academia" for excellence in research, funded by the Generalitat de Catalunya.

Appendix A. Variational integrator schemes

Appendix A.1. Midpoint rule with constant mass

We have seen that for PCA model reduction, the mass matrix is not position-dependent. The discrete Lagrangian for the generalized midpoint rule is then

$$\begin{aligned} L_d(q_0, q_1) &= \Delta t\, L\Big((1-\alpha)q_0 + \alpha q_1, \frac{q_1 - q_0}{\Delta t}\Big) \\ &= \Delta t \left\{ \frac{1}{2}\Big(\frac{q_1 - q_0}{\Delta t}\Big)^T \widetilde{M}\Big(\frac{q_1 - q_0}{\Delta t}\Big) - \widetilde{V}\Big((1-\alpha)q_0 + \alpha q_1\Big)\right\}, \end{aligned}$$

where $\alpha \in [0, 1]$. By making the discrete action stationary, $\frac{\partial S_d}{\partial q_k} = 0$, we find as discrete Euler–Lagrange equations

$$\begin{aligned} 0 = \frac{\partial S_d}{\partial q_k} \quad &= \quad \widetilde{M}\Big(\frac{q_k - q_{k-1}}{\Delta t}\Big) - \widetilde{M}\Big(\frac{q_{k+1} - q_k}{\Delta t}\Big) \\ &\quad -\Delta t\,(1-\alpha) D\widetilde{V}\Big((1-\alpha)q_k + \alpha q_{k+1}\Big) - \Delta t\,\alpha D\widetilde{V}\Big((1-\alpha)q_{k-1} + \alpha q_k\Big), \end{aligned}$$

which for $\alpha = 0$ gives the classical second-order Newmark explicit integration scheme

$$\widetilde{M}\Big(\frac{q_{k+1} - 2q_k + q_{k-1}}{\Delta t^2}\Big) = -D\widetilde{V}(q_k).$$

For PCA, the resulting $d \times d$ linear system of equations is

$$P^T M P\Big(\frac{q_{k+1} - 2q_k + q_{k-1}}{\Delta t^2}\Big) = -P^T DV(Pq_k + \bar{x}).$$

Appendix A.2. Midpoint rule with configuration-dependent mass and $\alpha = 0$

We consider a midpoint rule with $\alpha = 0$ to integrate the action between q_0 and q_1, which in the case of a mass configuration-dependent system generates the following discrete Lagrangian

$$\begin{aligned} L_d(q_0, q_1) &= \Delta t\, L\Big(q_0, \frac{q_1 - q_0}{\Delta t}\Big) \\ &= \Delta t \left\{ \frac{1}{2}\Big(\frac{q_1 - q_0}{\Delta t}\Big)^T \widetilde{M}(q_0)\Big(\frac{q_1 - q_0}{\Delta t}\Big) - \widetilde{V}(q_0)\right\}. \end{aligned}$$

The discrete Euler-Lagrange equations are then

$$\begin{aligned} 0 = \frac{\partial S_d}{\partial q_k} \quad &= \quad \Delta t \left\{ \frac{1}{2}\Big(\frac{q_{k+1} - q_k}{\Delta t}\Big)^T D\widetilde{M}(q_k)\Big(\frac{q_{k+1} - q_k}{\Delta t}\Big) - D\widetilde{V}(q_k)\right\} \\ &\quad -\widetilde{M}(q_k)\Big(\frac{q_{k+1} - q_k}{\Delta t}\Big) + \widetilde{M}(q_{k-1})\Big(\frac{q_k - q_{k-1}}{\Delta t}\Big). \end{aligned}$$

Appendix A.2.1. Generalized trapezoidal rule with configuration-dependent mass

The discrete Lagrangian for the generalized trapezoidal rule with configuration-dependent mass is

$$\begin{aligned} L_d(q_0, q_1) &= \Delta t \left\{ (1-\alpha)\, L\Big(q_0, \frac{q_1 - q_0}{\Delta t}\Big) + \alpha\, L\Big(q_1, \frac{q_1 - q_0}{\Delta t}\Big)\right\} \\ &= \Delta t \left\{ \frac{1}{2}\Big(\frac{q_1 - q_0}{\Delta t}\Big)^T \Big((1-\alpha)\,\widetilde{M}(q_0) + \alpha\,\widetilde{M}(q_1)\Big)\Big(\frac{q_1 - q_0}{\Delta t}\Big) - \Big((1-\alpha)\,\widetilde{V}(q_0) + \alpha\,\widetilde{V}(q_1)\Big)\right\}. \end{aligned}$$

The discrete Euler-Lagrange equations are then

$$
\begin{aligned}
0 = \frac{\partial S_d}{\partial q_k} &= \Delta t\,(1-\alpha)\left\{D_1L\left(q_k, \frac{q_{k+1}-q_k}{\Delta t}\right) + D_2L\left(q_k, \frac{q_{k+1}-q_k}{\Delta t}\right)\left(-\frac{1}{\Delta t}\right)\right\} \\
&\quad +\Delta t\left\{D_2L\left(q_{k+1}, \frac{q_{k+1}-q_k}{\Delta t}\right)\left(-\frac{\alpha}{\Delta t}\right) + D_2L\left(q_{k-1}, \frac{q_k-q_{k-1}}{\Delta t}\right)\frac{(1-\alpha)}{\Delta t}\right\} \\
&\quad +\Delta t\,\alpha\left\{D_1L\left(q_k, \frac{q_k-q_{k-1}}{\Delta t}\right) + D_2L\left(q_k, \frac{q_k-q_{k-1}}{\Delta t}\right)\frac{1}{\Delta t}\right\} \\
&= \Delta t\,(1-\alpha)\left\{\frac{1}{2}\left(\frac{q_{k+1}-q_k}{\Delta t}\right)^T D\widetilde{M}(q_k)\left(\frac{q_{k+1}-q_k}{\Delta t}\right) - D\widetilde{V}(q_k)\right\} \\
&\quad -\left((1-\alpha)\,\widetilde{M}(q_k) + \alpha\,\widetilde{M}(q_{k+1})\right)\left(\frac{q_{k+1}-q_k}{\Delta t}\right) \\
&\quad +\left((1-\alpha)\,\widetilde{M}(q_k) + \alpha\,\widetilde{M}(q_{k-1})\right)\left(\frac{q_k-q_{k-1}}{\Delta t}\right) \\
&\quad +\Delta t\,\alpha\left\{\frac{1}{2}\left(\frac{q_k-q_{k-1}}{\Delta t}\right)^T D\widetilde{M}(q_k)\left(\frac{q_k-q_{k-1}}{\Delta t}\right) - D\widetilde{V}(q_k)\right\}.
\end{aligned}
$$

Appendix B. Newton's method for the midpoint rule and $\alpha = 0$

We first define the residual of the discrete Euler–Lagrange equations as

$$r(q_{k+1}) = \frac{\partial S_d}{\partial q_k},$$

where $\partial S/\partial q_k$ is given by the Eq. (11). The first derivative of the residual is

$$J(q_{k+1}) = \frac{\partial r(q_{k+1})}{\partial q_{k+1}} = \left(\frac{q_{k+1}-q_k}{\Delta t}\right)^T D\widetilde{M}(q_k) - \frac{1}{\Delta t}\widetilde{M}(q_k).$$

The derivative of the reduced mass matrix involves second order derivatives of the parametrization $\mathbb{X}(q)$, see e.g. our work in thin shell analysis [8]. We numerically solve the discrete Euler-Lagrange equations in each time step by iterating

$$q_{k+1}^{i+1} = q_{k+1}^i - J^{-1}(q_{k+1}^i)\, r(q_{k+1}^i).$$

As an initial seed for Newton's method, we take that $D\widetilde{M}(q_k) = 0$, and then q_{k+1}^0 can be computed by solving the $d \times d$ system

$$\frac{\widetilde{M}(q_k)}{\Delta t}\left(\frac{q_{k+1}^0 - q_k}{\Delta t}\right) = \frac{\widetilde{M}(q_{k-1})}{\Delta t}\left(\frac{q_k - q_{k-1}}{\Delta t}\right) - D\widetilde{V}(q_k).$$

Appendix C. Nonlinear Locally Isometric Embedding

In this appendix, we describe an iterative procedure that finds a low-dimensional embedding as isometric as possible to the high-dimensional input. This is performed by minimization of a cost or stress function E_S, which measures the difference between the distances computed in low- and high-dimensions. This method is based on the work by [20], but differs in the way we measure distances in the ambient space, and in the stress energy. Here, we consider geodesic instead of Euclidean distances. The geodesic distance on the manifold is approximated as the shortest path distance in a graph made with $k-$nearest neighbors, in the same way that in the Isomap method [14].

We define the stress energy, to be minimized by the set of embedded configurations $Q = \{q_1, \ldots, q_N\} \in \mathbb{R}^d$, as

$$E_S(Q) = \alpha E_1(Q) + (1-\alpha)E_2(Q),$$

where

$$E_1(Q) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j\in N(i)}\left(\frac{|q_{ij}|}{|x_{ij}|} - 1\right)^2,$$

and

$$E_2(Q) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j\in N(i)}\left(\frac{|x_{ij}|}{|q_{ij}|} - 1\right)^2,$$

where $\alpha \in (0,1]$, $q_{ij} = q_i - q_j$, $x_{ij} = x_i - x_j$, and $\mathcal{N}(i)$ contains the indices for the first k_G nearest neighbors of x_i. The second term, E_2, prevents nearby points from colliding. From our experience, α should be close to 1. If the input data is very noisy in the normal direction to the manifold, a small α could lead to a very slow convergence, or even divergence. In such cases, we recommend $0.9999 \leq \alpha \leq 1$. In our experiments we commonly use $\alpha = 0.99 - 0.999$ for $d > 1$, and $\alpha = 1$ for $d = 1$.

In our definition of stress function, we take into account only a moderate number of neighboring points, which give an accurate idea of the local geometry of the manifold. Commonly, in our experiments we use $k_G = 200 - 1000$. Note that in the Isomap method, $k_G = N$. In the original work of Sammon [20], all the pairs distances are needed, which implies storing and manipulating a full matrix, whereas here we handle a sparse matrix. Similar methods have been proposed, for instance see the works [21, 22].

We numerically solve the nonlinear optimization problem of minimizing E_S by first iterating a L-BFGS algorithm [23] with a coarse tolerance, and then moving to Newton's method with a Brent's line search with a tighter tolerance. The gradient of the stress energy is

$$\frac{\partial E_S}{\partial q_i} = \alpha \frac{\partial E_1}{\partial q_i} + (1-\alpha)\frac{\partial E_2}{\partial q_i},$$

where

$$\frac{\partial E_1}{\partial q_i} = \sum_{j \in \mathcal{N}(i)} \left(\frac{|q_{ij}| - |x_{ij}|}{|x_{ij}|^2} \right) \frac{q_{ij}}{|q_{ij}|} + \sum_{j \in \widetilde{\mathcal{N}}(i)} \left(\frac{|q_{ij}| - |x_{ij}|}{|x_{ij}|^2} \right) \frac{q_{ij}}{|q_{ij}|},$$

$$\frac{\partial E_2}{\partial q_i} = \sum_{j \in \mathcal{N}(i)} \left(\frac{|q_{ij}| - |x_{ij}|}{|q_{ij}|^2} \right) \frac{q_{ij}}{|q_{ij}|} \frac{|x_{ij}|}{|q_{ij}|} + \sum_{j \in \widetilde{\mathcal{N}}(i)} \left(\frac{|q_{ij}| - |x_{ij}|}{|q_{ij}|^2} \right) \frac{q_{ij}}{|q_{ij}|} \frac{|x_{ij}|}{|q_{ij}|}.$$

The dual index list $\widetilde{\mathcal{N}}(i)$ to $\mathcal{N}(i)$ is built in such a way that $\widetilde{\mathcal{N}}(i)$ returns the indices j containing i in theirs index lists $\mathcal{N}(j)$. Finally, the Hessian of $E_S(q)$ is

$$\frac{\partial^2 E_S}{\partial q_i \partial q_j} = \alpha \frac{\partial^2 E_1}{\partial q_i \partial q_j} + (1-\alpha)\frac{\partial^2 E_2}{\partial q_i \partial q_j}.$$

The first term contributes diagonal terms

$$\begin{aligned} \frac{\partial^2 E_1}{\partial q_i^2} = & \sum_{j \in \mathcal{N}(i)} \left[\left(\frac{q_{ij}}{|q_{ij}|} \otimes \frac{q_{ij}}{|q_{ij}|} \right) \frac{|x_{ij}|}{|q_{ij}|} + \left(\frac{|q_{ij}| - |x_{ij}|}{|q_{ij}|} \right) \mathbb{I} \right] \frac{1}{|x_{ij}|^2} \\ & + \sum_{j \in \widetilde{\mathcal{N}}(i)} \left[\left(\frac{q_{ij}}{|q_{ij}|} \otimes \frac{q_{ij}}{|q_{ij}|} \right) \frac{|x_{ij}|}{|q_{ij}|} + \left(\frac{|q_{ij}| - |x_{ij}|}{|q_{ij}|} \right) \mathbb{I} \right] \frac{1}{|x_{ij}|^2}, \end{aligned}$$

and off-diagonal terms when j is in $\mathcal{N}(i)$ or $\widetilde{\mathcal{N}}(i)$, given by

$$\frac{\partial^2 E_1}{\partial q_i \partial q_j} = -\left[\left(\frac{q_{ij}}{|q_{ij}|} \otimes \frac{q_{ij}}{|q_{ij}|} \right) \frac{|x_{ij}|}{|q_{ij}|} + \left(\frac{|q_{ij}| - |x_{ij}|}{|q_{ij}|} \right) \mathbb{I} \right] \frac{1}{|x_{ij}|^2}.$$

From the second term of the stress energy, we have

$$\begin{aligned} \frac{\partial^2 E_2}{\partial q_i^2} = & \sum_{j \in \mathcal{N}(i)} \left[\left(\frac{q_{ij}}{|q_{ij}|} \otimes \frac{q_{ij}}{|q_{ij}|} \right) \frac{|x_{ij}|}{|q_{ij}|} + \left(\frac{|q_{ij}| - |x_{ij}|}{|q_{ij}|} \right) \left(\mathbb{I} - 3 \frac{q_{ij}}{|q_{ij}|} \otimes \frac{q_{ij}}{|q_{ij}|} \right) \right] \frac{|x_{ij}|}{|q_{ij}|^3} \\ & + \sum_{j \in \widetilde{\mathcal{N}}(i)} \left[\left(\frac{q_{ij}}{|q_{ij}|} \otimes \frac{q_{ij}}{|q_{ij}|} \right) \frac{|x_{ij}|}{|q_{ij}|} + \left(\frac{|q_{ij}| - |x_{ij}|}{|q_{ij}|} \right) \left(\mathbb{I} - 3 \frac{q_{ij}}{|q_{ij}|} \otimes \frac{q_{ij}}{|q_{ij}|} \right) \right] \frac{|x_{ij}|}{|q_{ij}|^3}, \end{aligned}$$

and

$$\frac{\partial^2 E_2}{\partial q_i \partial q_j} = -\left[\left(\frac{q_{ij}}{|q_{ij}|} \otimes \frac{q_{ij}}{|q_{ij}|} \right) \frac{|x_{ij}|}{|q_{ij}|} + \left(\frac{|q_{ij}| - |x_{ij}|}{|q_{ij}|} \right) \left(\mathbb{I} - 3 \frac{q_{ij}}{|q_{ij}|} \otimes \frac{q_{ij}}{|q_{ij}|} \right) \right] \frac{|x_{ij}|}{|q_{ij}|^3}.$$

References

[1] P. Krysl, S. Lall, J. E. Marsden, Dimensional model reduction in non-linear finite element dynamics of solids and structures, International Journal for Numerical Methods in Engineering 51 (4) (2001) 479–504.

[2] S. Niroomandi, I. Alfaro, E. Cueto, F. Chinesta, Real-time deformable models of non-linear tissues by model reduction techniques, Computer Methods and Programs in Biomedicine 91 (2008) 223–231.
[3] J. Lee, M. Verleysen, Nonlinear Dimensionality Reduction, Information science and statistics, Springer, New York, NY, USA, 2007.
[4] A. J. Gámez, C. S. Zhou, A. Timmermann, J. Kurths, Nonlinear dimensionality reduction in climate data, Nonlinear Processes in Geophysics 11 (2004) 393–398.
[5] P. Das, M. Moll, H. Stamati, L. Kavraki, C. Clementi, Low-dimensional, free-energy landscapes of protein-folding reactions by nonlinear dimensionality reduction, Proceedings of the National Academy of Sciences 103 (26) (2006) 9885–9890.
[6] W. M. Brown, S. Martin, S. N. Pollock, E. A. Coutsias, J.-P. Watson, Algorithmic dimensionality reduction for molecular structure analysis, The Journal of Chemical Physics 129 (6) (2008) 064118.
[7] J. Vanderplas, A. Connolly, Reducing the dimensionality of data: Locally linear embedding of sloan galaxy spectra, The Astronomical Journal 138 (5) (2009) 1365–1379.
[8] D. Millán, A. Rosolen, M. Arroyo, Nonlinear manifold learning for meshfree finite deformations thin shell analysis, International Journal for Numerical Methods in Engineering. Accepted.
[9] M. Arroyo, L. Heltai, D. Millán, A. DeSimone, Reverse engineering the euglenoid movement, Proceedings of the National Academy of Sciences , under review.
[10] M. Arroyo, M. Ortiz, Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods, International Journal for Numerical Methods in Engineering 65 (13) (2006) 2167–2202.
[11] F. Chinesta, A. Ammar, E. Cueto, Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models, Archives for Numerical Methods in Engineering 17 (2010) 327–350.
[12] D. Amsallem, C. Farhat, An online method for interpolating linear parametric reduced-order models, SIAM Journal of Scientific Computing 33 (2011) 2169–2198.
[13] K. Pearson, On lines and planes of closest fit to systems of points in space, Philosophical Magazine 2 (6) (1901) 559–572.
[14] J. Tenenbaum, V. de Silva, J. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (2000) 2319–2323.
[15] M. P. do Carmo, Differential geometry of curves and surfaces, Prentice-Hall, 1976.
[16] J. Marsden, M. West, Discrete mechanics and variational integrators, Acta Numerica 10 (2001) 357–514.
[17] A. Lew, J. E. Marsden, M. Ortiz, M. West, Variational time integrators, International Journal for Numerical Methods in Engineering 60 (1) (2004) 153–212.
[18] A. Rosolen, D. Millán, M. Arroyo, On the optimum support size in meshfree methods: a variational adaptivity approach with maximum entropy approximants, International Journal for Numerical Methods in Engineering 82 (7) (2010) 868–895.
[19] P. Español, Statistical Mechanics of Coarse-Graining, Lecture Notes in Physics, Springer-Verlag, 2003.
[20] J. W. Sammon, A nonlinear mapping for data structure analysis, IEEE Transactions on Computers 18 (1969) 401–409.
[21] J. Hérault, A. Oliva, A. Guérin-Dugué, Scene categorisation by curvilinear component analysis of low frequency spectra, in: M. Verleysen (Ed.), Proceedings of the 5th European Symposium on Artificial Neural Networks, Bruges, Belgium, 1997, pp. 91–96.
[22] J. Lee, A. Lendasseb, M. Verleysen, Nonlinear projection with curvilinear distances: Isomap versus curvilinear distance analysis, Neurocomputing 57 (2004) 49–76.
[23] J. Nocedal, S. Wright, Numerical Optimization, Springer, USA, 1999.

Appendix D

Paper # 4: "Reverse engineering the euglenoid movement"

Reverse engineering the euglenoid movement

Marino Arroyo [*], **Luca Heltai** [†], **Daniel Millán** [*] **and Antonio DeSimone** [†]

[*]Universitat Politècnica de Catalunya-BarcelonaTech, arcelona, Spain, and [†]SISSA-International School for Advanced Studies, Trieste, Italy

Submitted to Proceedings of the National Academy of Sciences of the United States of America

Euglenids exhibit an unconventional motility strategy amongst unicellular eukaryotes, consisting of large amplitude highly concerted deformations of the entire body (euglenoid movement or metaboly). A plastic cell envelope called pellicle mediates these deformations. Unlike ciliary or flagellar motility, the biophysics of this mode is not well understood, including its efficiency and molecular machinery. We examine quantitatively video recordings of four euglenids executing such motions with statistical learning methods. This analysis reveals strokes of high uniformity in shape and pace. We then interpret the observations in the light of a theory for the pellicle kinematics, providing a precise understanding of the link between local actuation by pellicle shear and shape control. We systematically understand common observations, such as the helical conformations of the pellicle, and identify previously unnoticed features of metaboly. While two of our euglenids execute their stroke at constant body volume, the other two exhibit deviations of about 20% from their average volume, challenging current models of low Reynolds number locomotion. We find that the active pellicle shear deformations causing shape changes can reach 340%, and estimate the velocity of the molecular motors. Moreover, we find that metaboly accomplishes locomotion at hydrodynamic efficiencies comparable to those of ciliates and flagellates. Our results suggest new quantitative experiments, provide insight into the evolutionary history of euglenids, and suggest that the pellicle may serve as a model for engineered active surfaces with applications in micro-fluidics.

low Reynolds number | microswimmers | stroke kinematics | locomotion | active soft matter

Unicellular microorganisms have developed effective ways of locomotion in a fluid, overcoming fundamental physical constraints such as the time reversibility of low Reynolds number (Re) hydrodynamics (1). Amongst eukaryotes, most species swim beating cilia or flagella. Yet, through a long evolutionary history, some protists have developed unconventional functional strategies, accomplished by highly diverse sub-cellular structures (2). A notable example is the euglenoid movement, or metaboly, executed by some species of euglenids (3). This peculiar motility mode is characterized by elegantly concerted, large amplitude distortions of the entire cell with frequencies of about $f \approx 0.1$ Hz (4). Euglenids have attracted the attention of scientists since the earliest days of microscopy, when van Leeuwenhoek referred to them in 1674 as microscopic motile "animalcules" that were green in the middle, which challenged the classification of organisms into animals and plants (5). Metaboly has inspired more recently models for artificial micro-swimmers (6), although even its locomotory function remains unclear. In contrast with flagellar or ciliary motion, the euglenoid movement has not undergone close biophysical scrutiny and fundamental questions remain open, including a precise understanding of the actuation mechanism leading to the body shape changes, or its hydrodynamic efficiency. Because all euglenids move primarily with their flagella, it is difficult to discern the role of the body distortions in the observed motion.

Euglenids are abundant in a wide range of aquatic environments, and with typical sizes from tens to hundreds μm, are easily observed by optical microscopy. They display a striking diversity in terms of morphology, nutrition (phagotrophs, phototrophs, osmotrophs), and motility, making of euglenids an excellent system to study evolution (7, 8). As other protists, they cannot resort to cell specialization to accomplish vital functions. Instead, they encapsulate all the functionality required for survival in a single cell, with an elaborate machinery revealed by ultrastructual studies. Euglenids are enclosed by a striated surface with two opposing poles called pellicle. The pellicle, the major trait of morphological diversity in the group, is a cortical complex consisting of the plasma membrane, a set of interlocking proteinaceous strips, microtubules, and tubular cisternae of endoplasmic reticulum arranged along the strips (3). The ability of euglenids to undergo metaboly has been correlated with the morphology of the pellicle. Flexible pellicles possess a large number of articulated strips (a few tens) often wrapped helically around the cell, whereas rigid pellicles can have as few as four fused strips arranged either helically or along the meridians of the cell. Some euglenids exploit their body flexibility to merely round up, elongate, or gently bend. Here we focus on the most extreme oscillatory motions displayed by some species.

Detailed observations of Euglena fusca, possessing a pellicle decorated by easily traceable particles, have shown that shape changes are accompanied by sliding between adjacent pellicle strips (9, 10); as the shape changes, the pellicle strips retain their length and width, but change their helicity. This mechanism is consistent with the commonly observed twisting motions around the long axis of the cell accompanying metaboly. These pellicle rearrangements are thought to be driven by the sliding of microtubule complexes relative to the proteinaceous strips, and controlled by calcium sequestration in the tubular cisternae of endoplasmic reticulum. Although the molecular motor has not been identified, the active motions are thought to be fueled by ATP (11). Further experimental interrogation of metaboly with modern biophysical techniques is still lacking.

The euglenoid movement confronts us with far-reaching biological and biophysical questions. One such issue is the competitive advantage of metaboly. Although metaboly is an oddity in microscopic locomotion, it has persisted through natural selection coexisting within individuals with flagellar motility in a significant part of the phylogenic tree of euglenids. Another intriguing aspect of metaboly is the operation of the distributed machine that elegantly and robustly executes the body motions, and that may inspire man-made active envelopes. As a first step, our goal here is to understand the kinematics and hydrodynamics of the euglenoid movement. For this purpose, we quantitatively analyze movies of metaboly recorded by others, representative of common observations by protistologists and physical scientists interested in motility (4). We examine these recordings with statistical learning methods. We then propose a model for the pellicle kinematics, which establishes a precise connection between distributed actuation and shape changes. Finally, we assess by numerical simulations the hydrodynamic features of the euglenoid movement.

Reserved for Publication Footnotes

Quantitative observations

Data and methods. We start from video recordings of four photosynthetic specimens, labelled euglenid #1 through #4, three of which are of the genus Eutreptiella, and the fourth belongs to either an Euglena or an Eutreptiella species. We assume that these euglenids remain essentially axisymmetric as they undergo metaboly, in agreement with their general description (12) and with detailed scrutiny of the movies studied here (see Fig. S1). Therefore, they can be described by their generating curve in the symmetry plane.

From the viewpoint of low Re locomotion, a swimming stroke is a closed path in shape space, which needs to be non-reciprocal to accomplish net motion. We propose next a method to identify and parametrize such path for further quantitative examination (see Fig. 1). In recent years, statistical learning methods have been increasingly used to understand stereotyped animal behavior, see, e.g., related studies on the motility of *C. elegans* (13, 14). Since the stroke is expected to be a nonlinear manifold of dimension one, here we resort to a nonlinear dimensionality reduction technique called Isomap (15). This method identifies nonlinear correlations of high-dimensional data points (here snapshots of the cells describing their shape) by embedding them in low dimensions in a geometry-preserving manner.

First (Fig. 1A), we gather the collection of shapes adopted by each euglenid by segmenting and aligning the video frames. We represent numerically the shape of each frame by fitting a spline curve to the boundary of the aligned segmentation. Second (Fig. 1B), we identify the geometric structure underlying the set of shapes (i) with Isomap. We embed the shapes in 2D (ii) to examine the nature of the path described by the euglenids in shape space. Since the shape ensemble is essentially a one-dimensional loop, we further embed the shapes in a periodic 1D segment labelled by τ (iii), where the frames are ordered by shape proximity. With the 1D embedding, we parametrize the geometric stroke smoothly (iv). This procedure filters some of the noise while closely following the data, see Fig. S2A,B. The resolution of the result is limited by optical artifacts and images out of focus, which bias the segmentation, and possibly by slight departures from axisymmetry. Yet, the quality of the numerical stroke is sufficient for quantitative examination (see Movie S1).

Because our method is based on geometric similarity, τ measures arc-length in shape space rather than physical time. The hydrodynamic efficiency depends on the relative rate at which different parts of the stoke are executed (the net displacement after one cycle does not). To inspect the pace along the stroke, we obtain from the data the relation between τ and normalized physical time t.

Results. We find that nonlinear dimensionality reduction reveals vividly the signature of low Re swimming for each of the four euglenids, with well-defined non-reciprocal paths in shape space over the many cycles captured in the movies (Figs. 2A and S3A). Different specimens perform geometrically distinct strokes (Fig. S2C), and, for a given specimen, each stroke is executed multiple times with the same pace, i.e., the relation between τ and t is nearly the same over several strokes (Figs. 2B and S3D). With the numerical stroke at hand, we first examine elementary features such as the surface area $S(t)$ and the volume $V(t)$ enclosed by the pellicle. We express volume in non dimensional terms as the reduced volume $v = 6\sqrt{\pi}V/S^{3/2}$, the ratio between the cell volume and the volume of a sphere with the same surface area. Thus, v ranges between 0 to 1. We find that the surface area of the four euglenids remains nearly constant along the stroke, with deviations below 5% (Fig. S2C). This observation is consistent with the hypothesis that pellicle deformation is mediated by relative sliding of pellicle strips, which retain their length and width (9). Such a deformation is called simple shear in continuum mechanics, and preserves area locally. The small area deviations may be due to the systematic errors in the data acquisition mentioned earlier, or to a slight area deformability of the pellicle. Strikingly, the body volume behaves very differently for euglenids #1 and #2 and for euglenids #3 and #4, see Fig. 2C. The former exhibit reduced volume deviations of around 20% and average reduced volumes of around 0.75, while the latter show essentially constant reduced volume of smaller magnitude. The volume changing euglenids necessarily flow significant amounts of fluid in and out of their body quite rapidly during the stroke. The contracting vacuole of euglenids, connected to an invagination called reservoir located at one end of the cell, is known to regulate water, although generally at longer time-scales. The motion of the internal organelles during metaboly, clearly seen in the movies, suggests that the volume variations are accommodated by a distributed fluid permeation rather than by localized jets.

Pellicle kinematics

Theory. The information about the kinematics of the pellicle obtained above is incomplete, since shape alone does not report on the motion of material particles; specifically, no quantitative data is available about the tangential motions within the symmetry plane and in the azimuthal direction. A careful inspection of some of the movies reveals that, while changing shape, the cell body twists non-uniformly around the symmetry axis. We describe next a theory for the pellicle kinematics, which allows us to recover the full velocity field of the pellicle, and also provides a precise understanding of the concerted actuation leading to metaboly.

The key assumption in this theory, as discussed earlier, is that the pellicle deforms by simple shear γ along the strips (Fig. 3A). Consequently, the pellicle deformation is locally area preserving, and we can label the material parallels of the axisymmetric pellicle with a body coordinate λ, measuring the fraction of total area delimited by a given parallel (see Fig. S4). By time-differentiation of the generating curve at fixed λ, we can compute the pellicle velocity within the symmetry plane.

We recover next the azimuthal motions and the pellicle shear. Consider an arbitrary but fixed conformation of the pellicle, given by its generating curve ($r_0(\lambda)$ and $z_0(\lambda)$) and by the orientation of the pellicle strips, which could be obtained by *in vivo* imaging. This orientation is given mathematically by the vector field $\boldsymbol{s}_0$ along the strips, or by the angle the strips form with the surface meridians $\alpha_0(\lambda)$ (Fig. 3A). We take this conformation as a reference. The strain of the pellicle surface relative to the reference configuration can be characterized at each point by the Cauchy-Green deformation tensor $\boldsymbol{C}$ (16), which by axisymmetry only depends on the body coordinate λ. From the actuation viewpoint, we obtain a first expression for $\boldsymbol{C}$, depending on the reference pellicle orientation and the pellicle shear, Fig. 3A. On the other hand, comparing the reference state with a new configuration, given by $r(\lambda)$, $z(\lambda)$, and the azimuthal displacement $\psi(\lambda)$, we obtain a second expression for $\boldsymbol{C}$, Fig. 3B. Matching the two expressions, we find the relation between the global conformation (including shape and pellicle helicity) and the local actuation, see *SI Text, section 2* for details and Fig. S5 for a validation.

Here, we have obtained from the videos a trajectory of shapes, $r(\lambda, t)$ and $z(\lambda, t)$. The theory then provides expressions for the associated azimuthal displacement $\psi(\lambda, t)$ and the pellicle shear $\gamma(\lambda, t)$, from which in turn we evaluate the full velocity field and the conformation of the deformed pellicle strips along the stroke. The theory can also be used to evaluate the deformed pellicle shape and helicity resulting from a prescribed pellicle shear $\gamma(\lambda, t)$ acting on a reference conformation, although we do not explore here the design of strokes.

The theory imposes a constraint on the shapes achievable from a given reference pellicle, namely $r/r_0 \geq |\sin \alpha_0|$. Consider a reference pellicle with strips parallel to the meridians, i.e. $\alpha_0(\lambda) = \pi/2$. Then the shape at instant t is accessible from the reference pellicle if $r(\lambda, t) \geq r_0(\lambda)$, consistent with the fact that thin parts of flexible pellicles display strips with almost no helicity. We can access

all the shapes of the stroke by defining $r_0(\lambda)$ as the smallest radius adopted by the material parallel λ, and integrate $z_0(\lambda)$ from local area preservation. This procedure defines an optimal reference pellicle for each euglenid, that achieves the observed shapes with minimal pellicle shear at every point; we adopt this reference pellicle in the absence of detailed imaging.

Results. Figure 4 summarizes the application of our theory to the four euglenids, highlighting the link between distributed actuation, pellicle conformation, and shape. The reconstructed pellicle shears $\gamma(\lambda, t)$ of the four euglenids exhibit distinct patterns, but also common features. A bump of high pellicle shear near the head of the cell (snapshot a) travels towards the tail (snapshots b and c). This traveling shear bump reflects in the shape as a bulge that moves from head to tail, reminiscent of peristaltic movements. When it reaches the tail, the shear bump decays while a new one appears at the head (snapshot d), before the cycle closes. Euglenid #1 exhibits the bluntest pattern and shape trajectory, and only reaches pellicle shears of 250%. In contrast, euglenid #4 executes a remarkably sharp and regular pattern of pellicle shears, reaching 340%, with a corresponding well-defined bulge that slides along a cylindrical body of nearly constant radius. Some species of osmotrophic genera, such as Astasia and Distigma, exhibit a similar swimming style to the photosynthetic euglenid #4. With an estimated pellicle width of $w \approx 250$ nm, we predict a sliding displacement between adjacent strips of up to $\delta \approx 850$ nm for this euglenid. By differentiating the map of $\gamma(\lambda, t)$ in Fig. 4, we find that $\dot{\gamma}$ reaches $20f$ Hz, leading to a maximum sliding velocity of $\dot{\delta} = 20fw \approx 500$ nm/s, consistent with the velocities of molecular motors along microtubules (17, 18). The similarity between our pellicle reconstructions and observations of flexible pellicles, with a clear correlation between radius and pellicle orientation, is remarkable.

Hydrodynamics of the stroke

Combining the movie processing and the theory for the pellicle kinematics, we compute the surface velocity during the strokes, and analyze the hydrodynamics by placing the model euglenids in a Newtonian fluid. With a typical size of 50 μm and a stroke frequency $f \approx 0.1$ Hz, we estimate Re $\approx 10^{-4}$. We solve the Stokes equations with a boundary integral method, and the linear velocity along z and the azimuthal spin of the cell are calculated from the self-propulsion condition (19), see *SI Text, section 3* for details. We assume that volume changes are accommodated by a uniform normal permeation velocity. The fluid velocity at the pellicle is then the surface velocity plus the permeation velocity, which is not very significant even for the volume changing euglenids #1 and #2 (see Fig. S5).

Table 1 summarizes the kinematic features and swimming performance of the four euglenids (see Table S1 for additional data). We evaluate the swimming performance with the linear displacement in one stroke U, measured in units of body length $2R = \sqrt{S/\pi}$, and with Lighthill's efficiency Eff_L. This conventional notion of efficiency is defined as the ratio between the power needed to drag the swimmer at the average velocity $W = U/T$ and the average power exerted by the swimmer on the surrounding fluid during the stroke, P^{out}. Since our swimmers change shape continuously, we consider $\mathrm{Eff}_L^{out} = 6\pi\eta RW^2/P^{out}$, where η is the fluid viscosity. We find that all euglenids move by a few tenths of a body length in one stroke, and exhibit efficiencies in the order of 1%. With the typical frequencies of metaboly, the swimming speeds are very low, $W \approx 1 - 2\,\mu$m/s, between one and two orders of magnitude slower than during flagellar locomotion. Remarkably, the hydrodynamic efficiency is comparable to that estimated for ciliates and flagellates (20). We find that euglenid #1 is the slowest and less efficient swimmer, followed by #2 and #3, with comparable performance, and finally #4 is the best, with a displacement of 40% of the body length per stroke and $\mathrm{Eff}_L^{out} = 2.0$ %. The table suggests that swimming performance is correlated with maximum pellicle shear. The best swimmer is a volume preserving specimen with the lowest reduced volume, although there is no clear trend in this respect. See Movie S2 for a visual depiction of the stoke kinematics and the resulting linear and angular motion for the four euglenids.

The hydrodynamics in the symmetry plane are completely decoupled from the azimuthal hydrodynamics, by the linearity and symmetry of the equations. Thus, the azimuthal energy dissipation in the fluid is a toll on the swimmer effficiency resulting from the mechanism used by the pellicle to change shape. It accounts for about 20% of the outer fluid dissipation (see Table S1). Lighthill's efficiency disregards the internal energy losses, passive or metabolic, known to be dominant in other microorganisms (21). Without attempting to evaluate these, we view now the swimmer as an isolated active surface embedded in a fluid, and consider the inner fluid dissipation as well. Remarkably, the inner dissipation is very large, about three times the outer counterpart, resulting in efficiencies about three times smaller.

Figure 5 illustrates the fluid flow patterns around euglenid #4, and correlates these with the forward displacement during the stroke. Similar results are obtained for the other euglenids. We can clearly distinguish between a power and a recovery phase. During the power stroke, a bulge slides down the slender body, producing a characteristic flow pattern that propels the swimmer forward monotonically. During the recovery stroke, the swimmer moves backwards and the bulge at the tail disappears at the expense of a growing bulge at the head. This requires a considerable forward rearrangement of fluid and pellicle mass, clearly noticeable in the flow pattern. The large internal streaming during this phase agrees with the motion of the organelles observed in movies of metabolic euglenids. While the energy dissipation in the outer fluid is quite uniformly distributed along the stroke for all four euglenids, the average dissipation power of the inner fluid is three to four times larger during the recovery phase than during the power phase. Since optimal strokes with respect to a dissipation metric exhibit uniform dissipation rate (22), this shows that the observed strokes do not minimize internal dissipation, despite it is the largest contribution. Figure 5B summarizes the correspondence between the distributed actuation mechanism and the resulting forward motion during the power and recovery phases.

Discussion

Our study suggests many new questions. A number of quantitative experiments could test and complement our theory. A detailed kinematical picture of the pellicle during metaboly can be obtained *in vivo* by light microscopy under ultraviolet illumination (23), while particle image velocimetry (24) can resolve the hydrodynamics and shed light on the inflow and outflow management of volume-changing euglenids. Our observation about the volume variations challenges the predominant conceptual models of swimming at low Re (25, 6), although swimmers permeating fluid across a fixed body shape have been analyzed (26). It is not clear if volume changes are passive, as a consequence of the pellicle deformation, or if these two phenomena are regulated in concert. Experiments on extracted cell models suggest that the euglenoid movement is mediated by a spatiotemporally non-uniform calcium sequestration, which triggers the pellicle activity (11). This mechanism could drive osmotically water permeation, which may enhance the exchange of nutrients and wastes. It is known that photosynthetic euglenids can acquire nutrients by osmotrophy.

Having established that metaboly is a competent motility mode, the evolutionary benefit of such a large biological investment remains unclear, when coexisting with flagella, an alternative strategy of demonstrated success. Current evolutionary theories suggest that the earliest ancestors, rigid and feeding on small bacteria, developed a flexible pellicle to engulf larger eukaryotic prey. Photosynthetic species evolved from phagotrophs after engulfing green algae, and some of these lost the pellicle flexibility by strip fusing (8). Presumably, the pellicle flexibility may have found a secondary utility,

besides mediating in eukaryovory, and may have persisted because of this in photosynthetic euglenids such as those studied here, as well as in some osmotrophs. The benefit of metaboly may be related to the stirring of the cytosol or the surrounding medium, or to locomotion. Interestingly, many euglenids live in interstitial, soil-rich, or eutrophic water, suggesting that the metaboly may be particularly well-suited to move in 'difficult' conditions such as confined environments, or in media of complex rheology, known to affect flagellar efficiency (20, 27). This hypothesis is supported by a marine photosynthetic species, lacking emergent flagella but exhibiting strong metaboly, known to migrate vertically through the sand of the benthic zone in diurnal cycles (28). Thus, the heterogeneity of the biophysical traits of metaboly could complement the morphological and molecular data used in current phylogenic studies of euglenids, and provide new clues about their evolutionary history.

Besides complementing experimental observations to obtain a complete kinematical picture of metaboly, the theory for the pellicle deformation provides the background for further biophysical studies. It is possible to optimize the pellicle shear to achieve maximal efficiency, and investigate the role of volume constraints or different dissipation metrics. Similar studies have brought new insight for cilia (29), flagella (30, 31), and model swimmers undergoing full body distortions (32, 22, 33). Furthermore, the theory can be used to compute the resisting forces, arising from the internal and external friction and from the pellicle bending elasticity, that need to overcome by metabolic activity.

The pellicle suggests itself as an appealing design concept for an active material (34). The local actuation of the pellicle produces a local strain incompatible with the current geometry, which is only realizable by further curving and producing shape changes (35). Similar concepts of material-as-machine relying on martensitic materials have been put forth (36, 37), and actually operate in biological organisms such as the tail sheath of bacteriophage T4 (38). Viewed as an active surface, the pellicle exhibits an interesting balance between shape flexibility and controllability, which explains why metaboly is such an elegantly concerted motion as compared, e.g., with the amoeboid movement. The pellicle has a single degree of freedom available for the local surface deformation, the pellicle shear, which nevertheless is sufficient to realize a wide family of shapes. Observations show that flexible euglenids can perform bending and torsional maneuvers of remarkable agility, including sharp turns. A general theory considering non-uniform pellicle shears in the azimuthal direction, $\gamma(u, \theta, t)$, may delimitate the possibilities of this shape actuation concept, which may find applications in artificial micro swimmers or peristaltic micro pumps.

Conclusions

We have provided a detailed biophysical analysis of the euglenoid movement, a widely appreciated but poorly understood motility mode of euglenids. We have developed a statistical learning methodology to analyze strokes and applied it to four movies recorded by others. We have established the shape and pace uniformity of the strokes, and have parametrized the motions in a computer model. Some euglenids exhibit very large variations of their body volume during the stroke, while others keep their volume constant. A kinematical theory, assuming that the cell surface deforms by simple shear along pellicle strips, provides a precise link between distributed actuation and shape changes, explains common observations about the pellicle helicity, and provides the input for hydrodynamics. Metaboly is found to be a slow but efficient motility mode in water, which enhances stirring of the cytosol and the surrounding fluid, and may be advantageous in granular or complex media.

ACKNOWLEDGMENTS. MA and DM acknowledge the support of the European Research Council (FP7/2007-2013)/ERC grant agreement nr 240487. MA acknowledges the support received through the prize "ICREA Academia" for excellence in research, funded by the Generalitat de Catalunya. We thank David Prat for assistance in processing some of the video frames, Brian S. Leander for helpful discussions, and Richard E. Triemer and Francisco Pujante for providing movies of euglenids.

1. Purcell E. M. (1977) Life at low Reynolds numbers. *Am J Phys* 45:3–11.
2. Lukes J, Leander B. S, Keeling P. J. (2009) Cascades of convergent evolution: The corresponding evolutionary histories of euglenozoans and dinoflagellates. *Proc Natl Acad Sci USA* 106:9963–9970.
3. Leander B. S. (2008) Euglenida. euglenids or euglenoids. Version 11 September 2008. http://tolweb.org/Euglenida/97461/2008.09.11 in The Tree of Life Web Project, http://tolweb.org/.
4. Fletcher D. A, Theriot J. A. (2004) An introduction to cell motility for the physical scientist. *Phys Biol* 1:T1–T10.
5. Dobell C. (1932) *Antony van Leeuwenhoek and his "Little Animals"*. (Dover, New York).
6. Avron J. E, Kenneth O, Oaknin D. H. (2005) Pushmepullyou: an efficient microswimmer. *New J Phys* 7:234.
7. Leander B. S, Witek R. P, Farmer M. A. (2001) Trends in the evolution of the euglenid pellicle. *Evolution* 55:2215–2235.
8. Leander B. S, Esson H. J, Breglia S. A. (2007) Macroevolution of complex cytoskeletal systems in euglenids. *BioEssays* 29:987–1000.
9. Suzaki T, Williamson R. E. (1985) Euglenoid movement in euglena fusca: Evidence for sliding between pellicular strips. *Protoplasma* 124:137–146.
10. Suzaki T, Williamson R. E. (1986) Cell surface displacement during euglenoid movement and its computer simulation. *Cell Motil Cytoskel* 6:186–192.
11. Suzaki T, Williamson R. E. (1986) Reactivation of the euglenoid movement and flagellar beating in detergent-extracted cells of astasia longa: different mechanisms of force generation are involved. *J Cell Sci* 80:75–89.
12. Triemer R. E. (1999) The euglenoid project, http://euglena.msu.edu/.
13. Stephens G. J, Johnson-Kerner B, Bialek W, Ryu W. S. (2008) Dimensionality and dynamics in the behavior of C. elegans. *PLoS Comput Biol* 4:e1000028.
14. Stephens G. J, Osborne L. C, Bialek W. (2011) Searching for simplicity in the analysis of neurons and behavior. *Proc Natl Acad Sci USA* 108:15565–15571.
15. Tenenbaum J. B, Silva V, Langford J. C. (2000) A global geometric framework for nonlinear dimensionality reduction. *Science* 290:2319–2323.
16. Holzapfel G. A. (2000) *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*. (Wiley).
17. Toba S, Watanabe T. M, Yamaguchi-Okimoto L, Toyoshima Y. Y, higuchi H. (2006) Overlapping hand-over-hand mechanism of single molecular motility of cytoplasmic dynein. *Proc Natl Acad Sci USA* 103:5741–5745.
18. Carter N. J, Cross R. A. (2005) Mechanics of the kinesin step. *Nature* 435:308–312.
19. Stone H. A, Samuel A. D. T. (1996) Propulsion of microorganisms by surface distortions. *Phys Rev Lett* 77:4102–4104.
20. Lauga E, Powers T. R. (2009) The hydrodynamics of swimming microorganisms. *Rep Prog Phys* 72:096601.
21. Katsu-Kimura Y, Nakaya F, Baba S. A, Mogami Y. (2009) Substantial energy expenditure for locomotion in ciliates verified by means of simultaneous measurement of oxygen consumption rate and swimming speed. *J Exp Biol* 212:1819–1824.
22. Alouges F, DeSimone A, Lefebvre A. (2008) Optimal strokes for low Reynolds number swimmers: an example. *J Nonlinear Sci* 18:277–302.
23. Gebeshuber I. C, et al. (2009) *Bio-Inspired Nanomaterials and Nanotechnology*, ed. Zhou Y. (Nova Science).
24. Drescher K, et al. (2009) Dancing volvox: Hydrodynamic bound states of swimming algae. *Phys Rev Lett* 102:168101.
25. Najafi A, Golestanian R. (2004) Simple swimmer at low Reynolds number: Three linked spheres. *Phys Rev E* 69:062901.
26. Spagnolie S. E, Lauga E. (2010) Jet propulsion without inertia. *Phys Fluids* 22:081902.
27. Liu B, Powers T. R, Breuer K. S. (2011) Force-free swimming of a model helical flagellum in viscoelastic fluids. *Proc Natl Acad Sci USA* 108:19516–19520.
28. Esson H. J, Leander B. S. (2008) Novel pellicle surface patterns on euglena obtusa (euglenophyta) from the marine benthic environment: Implications for pellicle development and evolution. *J Phycol* 44:132–141.
29. Osterman N, Vilfan A. (2011) Finding the ciliary beating pattern with optimal efficiency. *Proc Natl Acad Sci USA* 108:15727–15732.
30. Spagnolie S. E, Lauga E. (2010) The optimal elastic flagellum. *Phys Fluids* 22:031901.
31. Tam D, Hosoi A. E. (2011) Optimal feeding and swimming gaits of biflagellated organisms. *Proc Natl Acad Sci USA* 108:1001–1006.
32. Avron J. E, Gat O, Kenneth O. (2004) Optimal swimming at low reynolds numbers. *Phys Rev Lett* 93:186001.
33. Alouges F, DeSimone A, Heltai L. (2011) Numerical strategies for stroke optimization of axisymmetric microswimmers. *Math Mod Meth Appl S* pp. 361–387.
34. Fletcher D. A, Geissler P. L. (2009) Active biological materials. *Annu Rev Phys Chem* 60:469–486.

35. **Klein Y, Efrati E, Sharon E. (2007) Shaping of elastic sheets by prescription of non-euclidean metrics. *Science* 315:1116–1120.**
36. **Bhattacharya K, DeSimone A, Hane K. F, James R. D, Palmstrom C. (1999) Tents and tunnels on martensitic films. *Mat Sci Eng A-Struct* 273-275:685–689.**
37. **Bhattacharya K, James R. D. (2005) The material is the machine. *Science* 307:53–54.**
38. **Kostyuchenko V. A, et al. (2005) The tail structure of bacteriophage t4 and its mechanism of contraction. *Nat Struct Mol Biol* 12:810–813.**

Table 1. Summary of the kinematic features and swimming performance of the four strokes. U denotes the net displacement in one stroke, in units of body length $2R = \sqrt{S/\pi}$ (the diameter of a sphere with the surface area of the swimmer). Eff_L denotes the Lighthill efficiency, accounting for either the outer, or both the outer and inner fluids.

Eugl	v_{min} | $\bar{v}$ | v_{max}	γ_{max}	U	Eff_L^{out} (%)	$\mathrm{Eff}_L^{out+inn}$ (%)
# 1	0.58 | 0.76 | 0.95	2.5	0.16	0.71	0.24
# 2	0.60 | 0.72 | 0.87	3.0	0.27	1.6	0.57
# 3	0.61 | 0.63 | 0.65	3.0	0.25	1.6	0.48
# 4	0.50 | 0.51 | 0.53	3.4	0.40	2.0	0.76

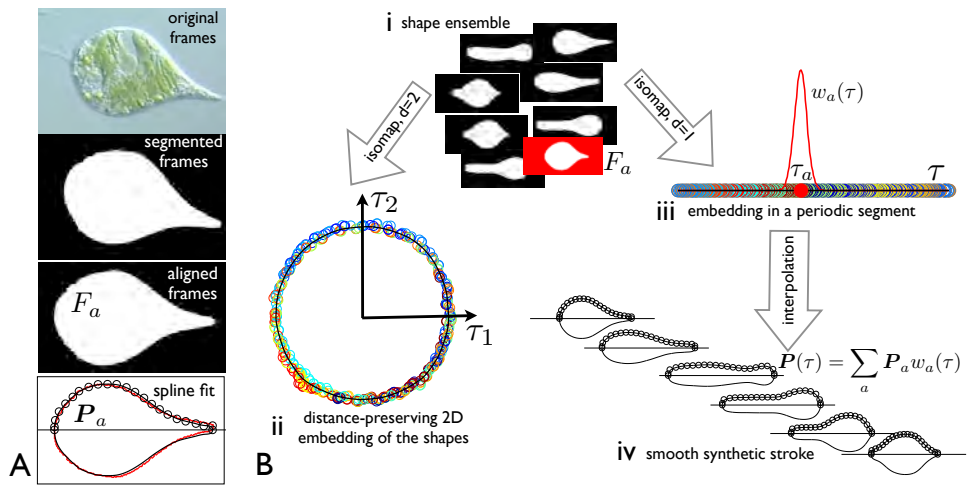


Fig. 1. Quantitative analysis of the movies: method. (A) The frames are segmented and aligned to obtain images F_a containing information about the shape alone, devoid of translation, rotation, or textures. A B-Spline curve, given by its control polygon $\boldsymbol{P}_a$ (black circles), is fitted to the boundary of F_a, and is a generating curve of the axisymmetric representation of the pellicle. (B) The segmented and aligned frames (i), representative of the shapes adopted by the cell, are embedded in low dimensions by a nonlinear dimensionality reduction technique called Isomap (15). The algorithm maps each frame to a low-dimensional point (circles in ii and iii, color labeling chronological order), so that the intrinsic distance between frames (shape dissimilarity) is preserved as much as possible by the low-dimensional representation. The embedding in the plane (τ_1, τ_2), (ii), shows that the stroke is a closed non-reciprocal path in shape space. Consequently, it can be most compactly described by embedding the frames in a periodic 1D segment (iii), from which we parametrize the stroke as a function of τ by interpolation (iv) with smooth basis functions $w_a(\tau)$. At any given τ, the synthetic stroke is a weighted average of the curves fitting video frames whose 1D embedding is in the vicinity of τ. The parameter τ is not proportional to physical time, ignored by the manifold learning algorithm, but rather to arc-length in shape space. See *SI Text, section 1* for technical details.

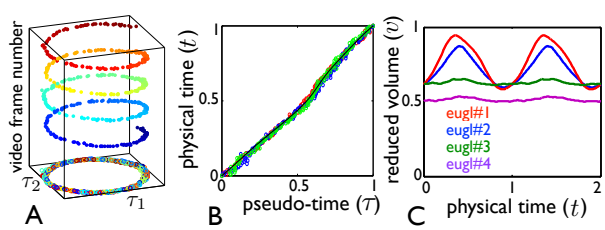


Fig. 2. Quantitative analysis of the movies: results. (A) Two-dimensional embedding of a stroke plotted against video frame number, showing the shape and pace uniformity of the stroke over several realizations. Isomap identifies similar shapes from different realizations (here about 4 cycles), and yields a single geometric stroke. (B) Pseudo-time parameter τ against normalized physical time t during three full strokes represented in different colors, showing that path in shape space is travelled with a well-defined pace (here, euglenid #3). The black line shows the fit used to re-parameterize time. (C) Reduced volume, v, as a function of time during two strokes. We distinguish two distinct motility styles: a volume-changing style for euglenids #1 and #2, and a volume-preserving style for euglenids #3 and #4. See Fig. S3 for the full data.

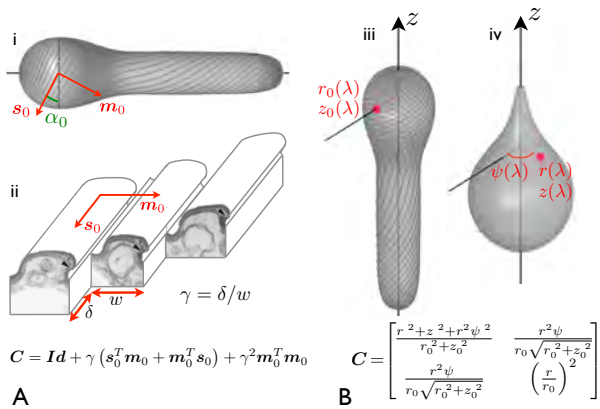


Fig. 3. Pellicle kinematics: theory. (A) Surface strain (the 2× 2 matrix $\boldsymbol{C}$) is derived assuming simple shear along the strips, γ, acting on a reference pellicle. The reference pellicle conformation (i,iii) is defined by a shape and a pellicle conformation, given by the tangent vector fields $\boldsymbol{s}_0$ and $\boldsymbol{m}_0$, along the strips and perpendicular to them, or by the angle α_0 between the pellicle strips and the surface parallels. (ii) Ultrastructure of the pellicle (transmission electron micrograph from (7)), and depiction of the sliding between adjacent strips causing the pellicle shear γ. (B) Surface strain, $\boldsymbol{C}$, is now derived by comparing a reference configuration, (iii), given by $r_0(\lambda)$ and $z_0(\lambda)$, where λ is the body coordinate, and a deformed configuration (iv), given by $r(\lambda)$, $z(\lambda)$ and the azimuthal displacement relative to a fixed direction $\psi(\lambda)$. The red point denotes a material particle attached the pellicle, and $(\,)'$ denotes differentiation with respect to λ. Matching the microstructural (A) and the shape-derived (B) expressions for the strain, we find equations relating pellicle shear, pellicle orientation, shape changes, and azimuthal motions.

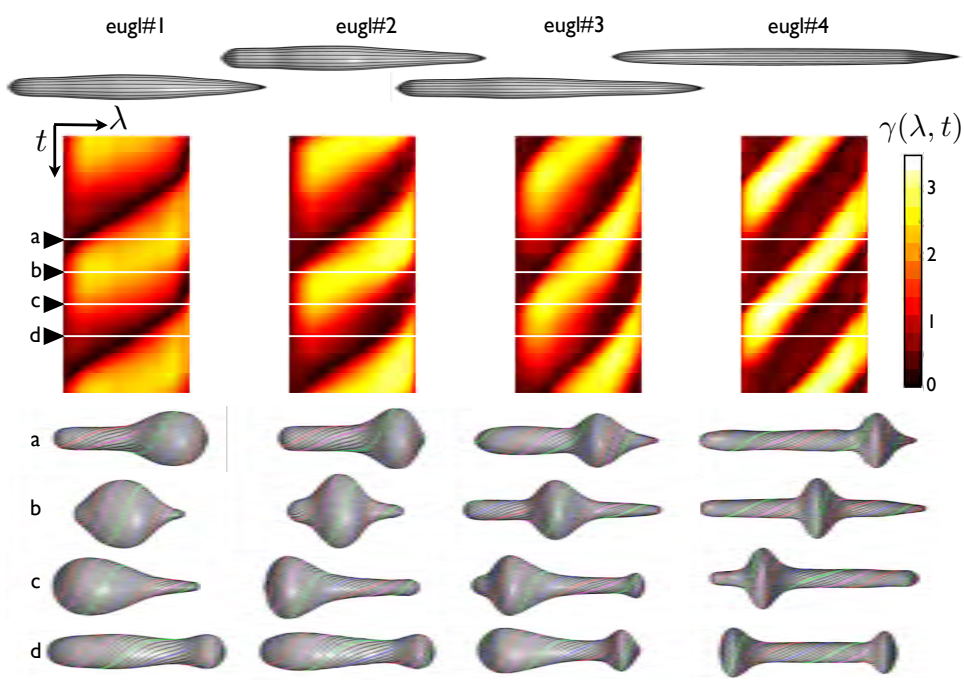


Fig. 4. Stroke kinematics: relationship between the local actuation and shape changes. Reference conformation of the pellicle for each euglenid (top), exhibiting very small reduced volumes. Map of the pellicle shear recovered from the observations as a function of time and body coordinate $\gamma(\lambda, t)$ (center), shown for two full synthetic strokes. The pellicle shear distributions acting on the reference configurations produce the shapes and pellicle conformations shown below.

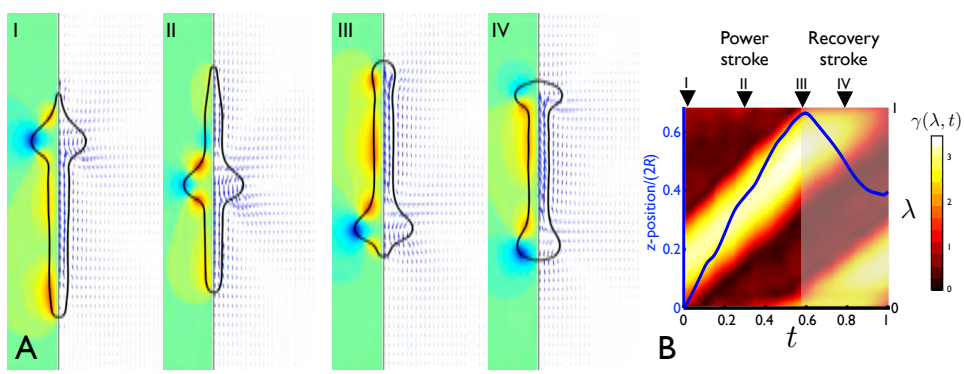


Fig. 5. Stroke hydrodynamics of euglenid #4. (A) Flow pattern around the pellicle at selected instants along the stroke. The fluid velocity field in the symmetry plane is indicated by blue arrows, and the azimuthal component is shown with iso-contours, where green is zero, blue negative and red positive. (B) Relationship between actuation and forward motion, where the vertical axis is the body coordinate λ for the color map of $\gamma(\lambda, t)$ and $z-$position (average between the head and the tail) in units of body length for the curve in blue. Both the actuation pattern and the forward motion clearly delineate a power and a recovery phase in the stroke.

Reverse engineering the euglenoid movement: Supporting Information

M. Arroyo, L. Heltai, D. Millán and A. DeSimone

August 14, 2012

S1 Video processing and parametrization of the stroke

The procedure outlined here assumes that the specimens adopt axisymmetric shapes, which agrees well with the detailed inspection of the videos, see Fig. S1. Additionally, the movies suggest that the axis of symmetry of the euglenid remains roughly parallel to the focal plane. Fully automated segmentation procedures require video inputs of high quality. The movies processed here include frames slightly out of focus, are non-uniformly illuminated, and have featured backgrounds. Consequently, our segmentation strongly relies on manual operations. We first enhance the edges of the cell by blurring the surroundings with Adobe Photoshop. This procedure also detaches the flagella from the cell body, and allows us to complete small body parts lying outside of the field of view. We then segment the resulting images with Matlab and ImageJ, and keep the best result. All other computations in the paper are carried out with in-house Matlab codes. The segmented frames are aligned (translated and rotated) with Principal Component Analysis (PCA). The boundary pixels are tagged, and split in two by the symmetry axis detected by PCA. One of the halves is reflected to yield a larger set of points sampling the axisymmetric surface. We then fit to these points a B-Spline curve [1] for each frame, expressed as

$$\boldsymbol{c}_a(u) = \sum_{I=1}^{N} B_I(u)\{r_I, z_I\}_a,$$

where $B_I(u)$ are the B-Spline basis functions referred to a reference interval $u \in [0,1]$ and $\{r_I, z_I\}_a$ are cylindrical coordinates of the control points associated with the $a-$th frame. We use cubic B-Splines with tangents along the r direction at the end points. The geometry can be adequately described with 20 knot intervals ($N = 23$ control points). However, for accurate resolution of the fluid flow, we consider 30 knot intervals (see Section S3). The video analysis is carefully checked frame by frame by visual inspection, and the procedure is iterated if needed. Movie S1 illustrates that this method provides a low-resolution, yet quantitative representation of the strokes.

We find low dimensional embeddings of the frames with Isomap [2], using 30 neighbors to approximate geodesic distances. The results are quite insensitive to this parameter in the range 20 to 50. This algorithm embeds each frame into a point τ_a in a periodic segment scaled to $[0,1]$. This segment represents a geometric pseudo-time. We use this embedding to parametrize in pseudo-time the generating curve of the euglenid surface during the stroke as

$$\boldsymbol{c}(u,\tau) = \sum_{a=1}^{n} w_a(\tau)\boldsymbol{c}_a(u) = \sum_{I=1}^{N} B_I(u) \sum_{a=1}^{n} w_a(\tau)\{r_I, z_I\}_a,$$

where n is the number of frames, and $w_a(\tau)$ are smooth meshfree maximum-entropy basis functions [3]. See Figure S2 for a discussion on the selection of their support size. Figure S3A shows the two-dimensional embeddings of the four euglenids, where each frame is mapped to a point $\boldsymbol{\tau}_a \in \mathbb{R}^2$, together with the reconstruction of the stroke path given by

$$\boldsymbol{\tau}(\tau) = \sum_{a=1}^{n} w_a(\tau)\boldsymbol{\tau}_a.$$

To estimate the relation between τ and physical time, we use the fact that the movies are shot at constant frame-rate Δt. For each frame associated to the time instant $(a-1)\Delta t$, we find the corresponding pseudo-instant by minimizing $|\boldsymbol{\tau}(\tau) - \boldsymbol{\tau}_a|^2$ with respect to τ. We then identify several distinct strokes and plot the resulting pairs of pseudo-time and physical time, normalized in the interval $[0,1]$ (see Figure S3D). We fit the one-to-one relation $t = g(\tau)$ with a spline shown in black, and reparameterize the reconstructed stroke in time as $\tilde{\boldsymbol{c}}(u,t) = \boldsymbol{c}\left(u, g^{-1}(t)\right)$, which in an abuse of notation we simply write as $\boldsymbol{c}(u,t)$.

S2 Pellicle kinematics

S2.1 Axisymmetric surface and generating curve

Let $\boldsymbol{c}(u,t) = \{r(u,t), z(u,t)\}$ denote the generating curve of the axisymmetric surface Γ acquired from the data as detailed above. We denote its speed by $a = \sqrt{r'^2 + z'^2}$, where $(\,)'$ denotes partial differentiation with respect to the space coordinate. A unit normal is given by $\boldsymbol{n} = 1/a\{-z', r'\}$. We orient the surface so that $\boldsymbol{n}$ is an outer normal. The element of area of the axisymmetric surface can be expressed as $dS = 2\pi a r\, du$. The surface area and volume of the axisymmetric surface can be computed as

$$S = 2\pi \int_0^1 ar\, du, \qquad V = \frac{1}{3}\int_\Gamma \boldsymbol{x}\cdot\boldsymbol{n}\, dS = \frac{2\pi}{3}\int_0^1 (-z'r + r'z)r\, du.$$

If the motion of the pellicle is locally area preserving with constant total area S, then the parameter

$$\lambda = f_t(u) = \frac{1}{S}\int_0^u 2\pi r(v,t)a(v,t)\, dv \tag{S1}$$

measuring the fraction of the total area delimited by u, labels material parallels of the pellicle surface during the motion. See Figure S4 for an illustration. By reparameterizing the generating curves with this body coordinate, we obtain a Lagrangian description (i.e., tracking material particles) of the pellicle up to azimuthal motions, $\hat{\boldsymbol{c}}(\lambda,t) = \{\hat{r}(\lambda,t), \hat{z}(\lambda,t)\} = \boldsymbol{c}(f_t^{-1}(\lambda),t)$. We assume from this point on that the surface is parametrized by λ, and with an abuse of notation, we denote the reparameterized curve by $\boldsymbol{c}(\lambda,t) = \{r(\lambda,t), z(\lambda,t)\}$.

S2.2 Deformation of the surface

The reference pellicle configuration is given by

$$\boldsymbol{x}_0(\lambda,\theta) = \{r_0(\lambda)\cos\theta, r_0(\lambda)\sin\theta, z_0(\lambda)\}\,, \tag{S2}$$

where the body coordinate $\lambda \in [0,1]$, and the azimuthal angle $\theta \in [0, 2\pi]$ label material particles. The full prescription of the pellicle Lagrangian motion along the stroke is given by the known functions $r(\lambda,t)$ and $z(\lambda,t)$, together with the unknown azimuthal displacement $\psi(\lambda,t)$

$$\boldsymbol{x}(\lambda,\theta,t) = \left\{ \begin{array}{c} r(\lambda,t)\cos[\theta+\psi(\lambda,t)] \\ r(\lambda,t)\sin[\theta+\psi(\lambda,t)] \\ z(\lambda,t) \end{array} \right\}. \tag{S3}$$

The natural basis vectors of the tangent plane to the deformed configuration can be expressed in Cartesian coordinates as [4]

$$\boldsymbol{x}_\lambda = \left\{ \begin{array}{c} r'\cos(\theta+\psi) - r\psi'\sin(\theta+\psi) \\ r'\sin(\theta+\psi) + r\psi'\cos(\theta+\psi) \\ z' \end{array} \right\}, \qquad \boldsymbol{x}_\theta = \left\{ \begin{array}{c} -r\sin(\theta+\psi) \\ r\cos(\theta+\psi) \\ 0 \end{array} \right\}.$$

By axisymmetry, the local geometry does not depend on the azimuthal position, and therefore without loss of generality we can particularize the expressions at $\theta+\psi=0$. Consequently, the metric tensor in the natural basis, $g_{ij} = \boldsymbol{x}_i \cdot \boldsymbol{x}_j$, takes the form

$$\boldsymbol{g} = \left[\begin{array}{cc} a^2 + r^2(\psi')^2 & r^2\psi' \\ r^2\psi' & r^2 \end{array} \right].$$

The components of this tensor are also those of the Cauchy-Green deformation tensor $\boldsymbol{C}$ expressed in the natural basis of the reference configuration given by $\boldsymbol{x}_{0\lambda}$ and $\boldsymbol{x}_{0\theta}$ [5]. Again, particularizing the expressions at $\theta = 0$ without loss of generality, these vectors can be expressed as

$$\boldsymbol{x}_{0\lambda} = \left\{ \begin{array}{c} r_0' \\ 0 \\ z_0' \end{array} \right\} = a_0 \boldsymbol{e}_{0\lambda}, \qquad \boldsymbol{x}_{0\theta} = \left\{ \begin{array}{c} 0 \\ r_0 \\ 0 \end{array} \right\} = r_0 \boldsymbol{e}_{0\theta},$$

where $\boldsymbol{e}_{0\lambda}$ and $\boldsymbol{e}_{0\theta}$ denote tangent unit vectors along meridians and parallels of the axisymmetric surface. By a change of basis, we find the components of $\boldsymbol{C}$ in the orthonormal basis $\{\boldsymbol{e}_{0\lambda}, \boldsymbol{e}_{0\theta}\}$, see Fig. 3B in the main text.

S2.3 Pellicle deformation

The tangent vector fields along and perpendicular to the strips can be expressed as

$$\boldsymbol{s}_0 = \sin\alpha_0 \boldsymbol{e}_{0\lambda} + \cos\alpha_0 \boldsymbol{e}_{0\theta}, \quad \boldsymbol{m}_0 = -\cos\alpha_0 \boldsymbol{e}_{0\lambda} + \sin\alpha_0 \boldsymbol{e}_{0\theta},$$

where $\alpha_0(\lambda)$ is the angle of the pellicle strips with the surface parallels in the reference configuration. Under the assumption that the local deformation is the result of simple shear along the pellicle strips, the local deformation gradient takes the form $\boldsymbol{F} = \boldsymbol{R}(\boldsymbol{Id} + \gamma \boldsymbol{s}_0 \otimes \boldsymbol{m}_0)$, where $\boldsymbol{R}$ is an orthogonal matrix. This local deformation is area preserving by construction. The Cauchy-Green deformation tensor takes the form

$$\begin{aligned} \boldsymbol{C} &= \boldsymbol{F}^T\boldsymbol{F} = \boldsymbol{Id} + \gamma(\boldsymbol{s}_0 \otimes \boldsymbol{m}_0 + \boldsymbol{m}_0 \otimes \boldsymbol{s}_0) + \gamma^2 \boldsymbol{m}_0 \otimes \boldsymbol{m}_0 \\ &= \left[\begin{array}{cc} 1 - 2\gamma\sin\alpha_0\cos\alpha_0 + \gamma^2\cos^2\alpha_0 & \gamma\left(\sin^2\alpha_0 - \cos^2\alpha_0\right) - \gamma^2\sin\alpha_0\cos\alpha_0 \\ \gamma\left(\sin^2\alpha_0 - \cos^2\alpha_0\right) - \gamma^2\sin\alpha_0\cos\alpha_0 & 1 + 2\gamma\sin\alpha_0\cos\alpha_0 + \gamma^2\sin^2\alpha_0 \end{array} \right]. \end{aligned}$$

By equating this microstructural deformation tensor (Fig. 3A) with the deformation derived from the shape changes (Fig. 3B), we obtain three equations

$$\left(a^2 + r^2\psi'^2\right)/a_0^2 = 1 - 2\gamma_0 \sin\alpha_0 \cos\alpha_0 + \gamma_0^2 \cos^2\alpha_0 \tag{S4}$$

$$\left(r^2\psi'\right)/(r_0 a_0) = \gamma_0\left(\sin^2\alpha_0 - \cos^2\alpha_0\right) - \gamma_0^2 \sin\alpha_0 \cos\alpha_0 \tag{S5}$$

$$(r/r_0)^2 = 1 + 2\gamma_0 \sin\alpha_0 \cos\alpha_0 + \gamma_0^2 \sin^2\alpha_0 \tag{S6}$$

This set of algebraic and differential equations relates the reference pellicle conformation (r_0, z_0, α_0), the deformed configuration (r, z, ψ), and the pellicle shear relative to the reference state (γ).

Assuming (r_0, z_0, α_0) and (r, z) are data, the equations above allow us to determine ψ and γ. The pellical shear follows from Eq. S6

$$\sin^2\alpha_0\gamma^2 + 2\sin\alpha_0\cos\alpha_0\gamma + 1 - (r/r_0)^2 = 0,$$

which has real solutions if the discriminant $\Delta = 4\sin^2\alpha_0\left[(r/r_0)^2 - \sin^2\alpha_0\right]$ is nonnegative, i.e., if

$$r/r_0 \geq |\sin\alpha_0|.$$

If $\Delta \geq 0$, the pellicle shear is

$$\gamma = \left(-\cos\alpha_0 \pm \sqrt{(r/r_0)^2 - \sin^2\alpha_0}\right)/\sin\alpha_0 \tag{S7}$$

The solution with a plus sign is the one with physical meaning, since the pellicle shear vanishes when the deformed and the reference configurations coincide. Then, Eq. S5 provides an ordinary differential equation (ODE) for ψ, or alternatively, combining Eqs. S5 and S6, we find

$$\frac{r^2}{a_0 r_0}\psi' = \frac{\cos\alpha_0}{\sin\alpha_0}\left[1 - \left(\frac{r}{r_0}\right)^2\right] + \gamma_0, \tag{S8}$$

which is easily integrated to find ψ. The initial condition for the ODE sets the surface rotation around the symmetry axis, and is determined by the hydrodynamics. For definiteness, we choose $\psi(0)$ such that $\int_\Gamma \psi r\, dS = 0$. By a direct calculation, plugging Eqs. S7 and S8 into Eq. S4, we find $\left[(ar)^2 - (a_0 r_0)^2\right]/(a_0 r)^2 = 0$, a compatibility condition identically satisfied if the motion is area preserving.

If a reference configuration with pellicle strips parallel to the meridians is considered, then the pellicle shear is $\gamma = \sqrt{(r/r_0)^2 - 1}$. This expression clearly shows that the larger r_0, the smaller γ. This is why the choice of reference pellicle given in the main text minimizes the pellicle shear point-wise.

S2.4 Validation of the theory

We use the experimental observations in [6] to examine the validity of the proposed theory, see Fig. S3. In this reference, two images showing different pellicle conformations of a specimen of Euglena fusca are reported. This species has a pellicle decorated with particles, making it easy to capture the pellicle conformation with light microscopy. We consider the left image as the reference conformation, and to simplify the analysis assume its pellicle strips are exactly meridians of the surface. We acquire from the image on the right r and z, and compute the pellicle shear and the azimuthal displacement required to accommodate this shape change. The resulting pellicle conformation agrees very well with the observation. This deformation mobilizes a maximum pellicle shear of 150%, which means that pellicle strips slide relative to each other as much as 1.5 times their width.

S2.5 Definition of a canonical pellicle conformation

By requiring that $\alpha_0 = \pi/2$, the pellicle equations are solvable along the stroke if $r(\lambda, t) \geq r_0(\lambda)$ for all $t \in [0, 1)$. We define $r_0(\lambda) = \min_t r(\lambda, t)$. To find $z_0(\lambda)$, we recall the definition of λ to write

$$\lambda S = 2\pi \int_0^\lambda \sqrt{{r_0'}^2(\nu) + {z_0'}^2(\nu)} r_0(\nu)\, d\nu.$$

Differentiating this expression with respect to λ, we find

$$z_0'(\lambda) = \sqrt{\left(\frac{S}{2\pi r_0(\lambda)}\right)^2 - r_0'(\lambda)}, \tag{S9}$$

which is easily integrated to find $z_0(\lambda)$.

S2.6 Pellicle velocity

Recalling Eq. S3 and the axisymmetry of the kinematics, the velocity field of the pellicle in the $x - z$ plane is

$$\dot{\boldsymbol{x}} = \left\{ \begin{array}{c} \dot{r} \\ r\dot{\psi} \\ \dot{z} \end{array} \right\} = v_n^s \boldsymbol{n} + v_\lambda \boldsymbol{e}_\lambda + v_\theta \boldsymbol{e}_\theta,$$

where the normal velocity due to shape changes given by $v_n^s = \boldsymbol{n} \cdot \dot{\boldsymbol{x}} = \left(-z'\dot{r} + r'\dot{z}\right)/a$, the tangential velocity along generating curves (meridians) is given by $v_\lambda = \boldsymbol{e}_\lambda \cdot \dot{\boldsymbol{x}} = \left(r'\dot{r} + z'\dot{z}\right)/a$ with $\boldsymbol{e}_\lambda = 1/a\{r', 0, z'\}$, and the tangential velocity along the parallels is given by $v_\theta = \boldsymbol{e}_\theta \cdot \dot{\boldsymbol{x}} = r\dot{\psi}$ with $\boldsymbol{e}_\theta = \{0, 1, 0\}$. By $\dot{(\,)}$ we denote differentiation with respect to time. See Figure S6 for an illustration.

S3 Hydrodynamics

For the volume changing euglenids, $\dot{V} = \int_\Gamma v_n^s\, dS \neq 0$. In order to satisfy the balance of mass of the internal fluid, assumed to be incompressible, the total normal fluid velocity needs to include a permeation contribution v_n^p satisfying $\int_\Gamma v_n^p\, dS = -\dot{V}$. The total fluid velocity at the pellicle surface is then

$$\boldsymbol{v}^{out} = (v_n^s + v_n^p)\boldsymbol{n} + v_\lambda \boldsymbol{e}_\lambda + (v_\theta + \omega r)\boldsymbol{e}_\theta + V_c \boldsymbol{e}_z,$$

where ω and V_c are (constant and a-priori unknown) rigid body rotational and translational velocities. The simplest possibility for the permeation normal velocity is a uniform field given by $v_n^p = -\dot{V}/S$, but any choice of the kind $v_n^p(u, t) = -\dot{V}h(u, t)$, where $h(u, t)$ is a profile with unit integral over the surface, is possible.

The interior part of the swimmer can be either ignored, or assumed to be a fluid with the same characteristics of the outer fluid. In this second option, the swimmer is seen as an isolated active surface embedded in a fluid, and the interior fluid satisfies Stokes equations with Dirichlet data given by $\boldsymbol{v}^{inn} = \boldsymbol{v}^{out}$.

We use a boundary integral formulation for the low Reynolds number (Re) hydrodynamics of a swimmer immersed in a quiescent (Newtonian) fluid [7]. Given a swimmer occupying the region Ω

with boundary Γ, the velocity of the fluid at any point $\boldsymbol{x}$ in space satisfies the equations

$$v_i(\boldsymbol{x}) + \int_\Gamma T_{ijk}(\boldsymbol{x}-\boldsymbol{y}) v_j^{inn}(\boldsymbol{y}) n_k(\boldsymbol{y}) d\Gamma_y = \int_\Gamma G_{ij}(\boldsymbol{x}-\boldsymbol{y}) f_j^{inn}(\boldsymbol{y}) d\Gamma_y \qquad \boldsymbol{x} \in \Omega \tag{S10}$$

$$v_i(\boldsymbol{x}) - \int_\Gamma T_{ijk}(\boldsymbol{x}-\boldsymbol{y}) v_j^{out}(\boldsymbol{y}) n_k(\boldsymbol{y}) d\Gamma_y = \int_\Gamma G_{ij}(\boldsymbol{x}-\boldsymbol{y}) f_j^{out}(\boldsymbol{y}) d\Gamma_y \qquad \boldsymbol{x} \in R^3 \setminus \Omega, \tag{S11}$$

where summation is implied over repeated indices, $\boldsymbol{v}$ is the velocity in the fluid, $\boldsymbol{f}^{inn}(\boldsymbol{y})$ and $\boldsymbol{f}^{out}(\boldsymbol{y})$ are the inner and outer tractions at $\boldsymbol{y} \in \Gamma$ defined as $\boldsymbol{f}^{inn}(\boldsymbol{y}) = \boldsymbol{\sigma}^{inn}(\boldsymbol{y})\boldsymbol{n}(\boldsymbol{y})$ and $\boldsymbol{f}^{out}(\boldsymbol{y}) = -\boldsymbol{\sigma}^{out}(\boldsymbol{y})\boldsymbol{n}(\boldsymbol{y})$, where $\boldsymbol{n}(\boldsymbol{y})$ is the outer unit normal to $\partial\Omega$ at the point $\boldsymbol{y}$, $\boldsymbol{\sigma}$ is the Cauchy stress in the fluid and

$$G_{ij}(\boldsymbol{r}) = \frac{1}{8\pi\nu}\left(\frac{\delta_{ij}}{|\boldsymbol{r}|} + \frac{r_i r_j}{|\boldsymbol{r}|^3}\right) \qquad T_{ijk}(\boldsymbol{r}) = -\frac{6}{8\pi}\left(\frac{r_i r_j r_k}{|\boldsymbol{r}|^5}\right)$$

are the free space Green functions of Stokes flow. Letting $\boldsymbol{x}$ approach Γ in (S10)-(S11) above, considering the integrals in the principal value sense and exploiting the symmetry of the domain we obtain two Fredholm integral equations of the first kind for the unknown tractions $\boldsymbol{f}^{inn}$ and $\boldsymbol{f}^{out}$, which we solve numerically via a Galerkin boundary element method [8].

Ignoring the inner behavior of the swimmer is equivalent to setting the Dirichlet boundary conditions for the inner velocity $\boldsymbol{v}^{inn}$ to zero. In this case the inner traction $\boldsymbol{f}^{inn}$ is also identically zero, and the interior part of the swimmer does not contribute to the motion, nor to the dissipation.

The a-priori unknown rigid body translational and rotational velocities V_c and ω of the swimmer are obtained by imposing the conditions of self-propulsion, which consistently with the low Re regime considered here, reduce to annihilating the total viscous force and torque. By axisymmetry, the only two non trivial equations are

$$\boldsymbol{e}_z \cdot \int_\Gamma (\boldsymbol{f}^{inn}(\boldsymbol{y}) + \boldsymbol{f}^{out}(\boldsymbol{y})) d\Gamma_y = 0, \qquad \boldsymbol{e}_z \cdot \int_\Gamma (\boldsymbol{y}-\boldsymbol{o}) \times (\boldsymbol{f}^{inn}(\boldsymbol{y}) + \boldsymbol{f}^{out}(\boldsymbol{y})) d\Gamma_y = 0.$$

where $\boldsymbol{o}$ is a point on the axis of symmetry of the swimmer. The power dissipated through the flow induced in the inner and outer fluid are given respectively by

$$W^{inn}(t) = \int_\Gamma \boldsymbol{f}^{inn}(\boldsymbol{y},t) \cdot \boldsymbol{v}(\boldsymbol{y},t) d\Gamma_y, \qquad W^{out}(t) = \int_\Gamma \boldsymbol{f}^{out}(\boldsymbol{y},t) \cdot \boldsymbol{v}(\boldsymbol{y},t) d\Gamma_y,$$

from which we obtain the average power exerted by the swimmer on the surrounding fluid P^{out} and the average ambient fluid dissipation $P^{int+out}$ when considering the swimmer as an isolated active surface embedded in the fluid:

$$P^{out} = \frac{1}{T}\int_0^T W^{out}(t) dt, \qquad P^{inn+out} = \frac{1}{T}\int_0^T (W^{inn}(t) + W^{out}(t)) dt.$$

Here T denotes the duration of one stroke.

References

[1] Piegl L, Tiller W. (1997) *The NURBS Book*. (Springer).

[2] Tenenbaum J. B, Silva V, Langford J. C. (2000) A global geometric framework for nonlinear dimensionality reduction. *Science* **290**:2319–2323.

[3] Arroyo M, Ortiz M. (2006) Local maximum-entropy approximation schemes: a seamless bridge between finite elements and meshfree methods. *Int J Numer Meth Eng* **65**:2167–2202.

[4] do Carmo M. (1976) *Differential geometry of curves and surfaces.* (Prentice-Hall).

[5] Marsden J, Hughes T. (1983) *The mathematical foundations of elasticity.* (Prentice-Hall).

[6] Suzaki T, Williamson R. E. (1985) Euglenoid movement in euglena fusca: Evidence for sliding between pellicular strips. *Protoplasma* **124**:137–146.

[7] Pozrikidis C. (1992) *Boundary integral and singularity methods for linearized viscous flow*, Cambridge Texts in Applied Mathematics. (Cambridge University Press, Cambridge).

[8] Alouges F, DeSimone A, Heltai L. (2011) Numerical strategies for stroke optimization of axisymmetric microswimmers. *Math Mod Meth Appl S* pp. 361–387.

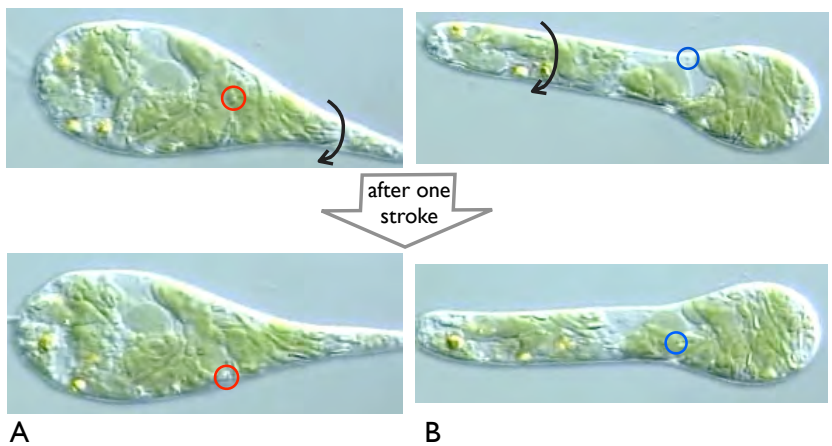


Figure S1: Evidence for near-axisymmetric cell shape. In some movies, particles attached to the pellicle are clearly noticeable. By tracking these markers and comparing snapshots one stroke apart, (A) and (B), we conclude that the cell has rotated about its axis by a significant amount. If the cell body was not axisymmetric, the shapes would differ significantly, and Isomap would show distinct paths for each realization of the stroke. However, this is not the case, see Fig. S3A.

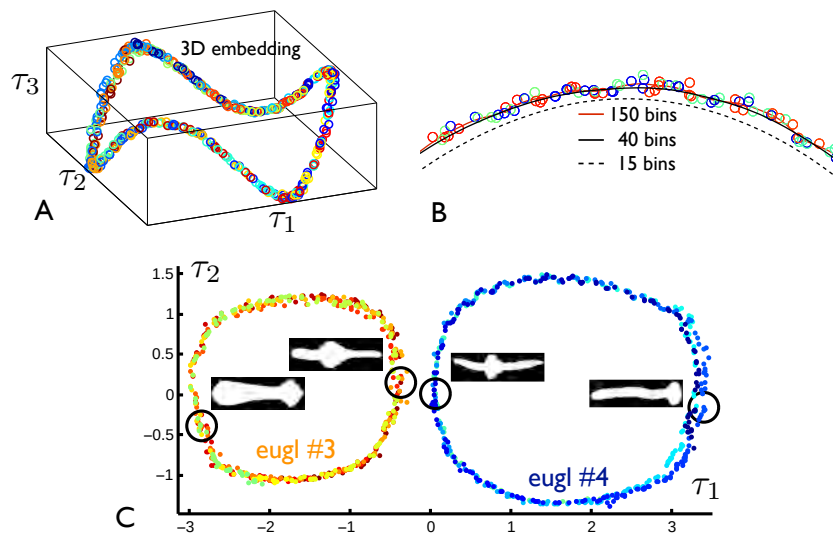


Figure S2: (A) Embedding of the frames in 3D for euglenid #3, showing that the specimen indeed follows a closed curve in shape space, and that the 2D picture is not the projection of a more complex trajectory. (B) Effect of the width of the basis functions on the reconstructed stroke, here visualized on the 2D embedding. The maximum entropy basis functions [3] used here satisfy $w_a(\tau) \propto \exp\left[-(n\tau)^2\right]$, where n is the number of bins used to subdivide the interval $[0, 1)$ and the typical support size of the basis functions is $\sim 1/n$. If n is too large, the reconstructed stroke follows spurious details caused by errors in the data acquisition, whereas if n is too small, then the stroke is excessively smoothed out and shrunk, resulting in less sharply reconstructed motions. We find an optimal compromise with $n = 40$. (C) Joint embedding of the volume-preserving euglenids #3 and #4, showing that the strokes are geometrically distinct, with nearby shapes in the middle of the power stroke.

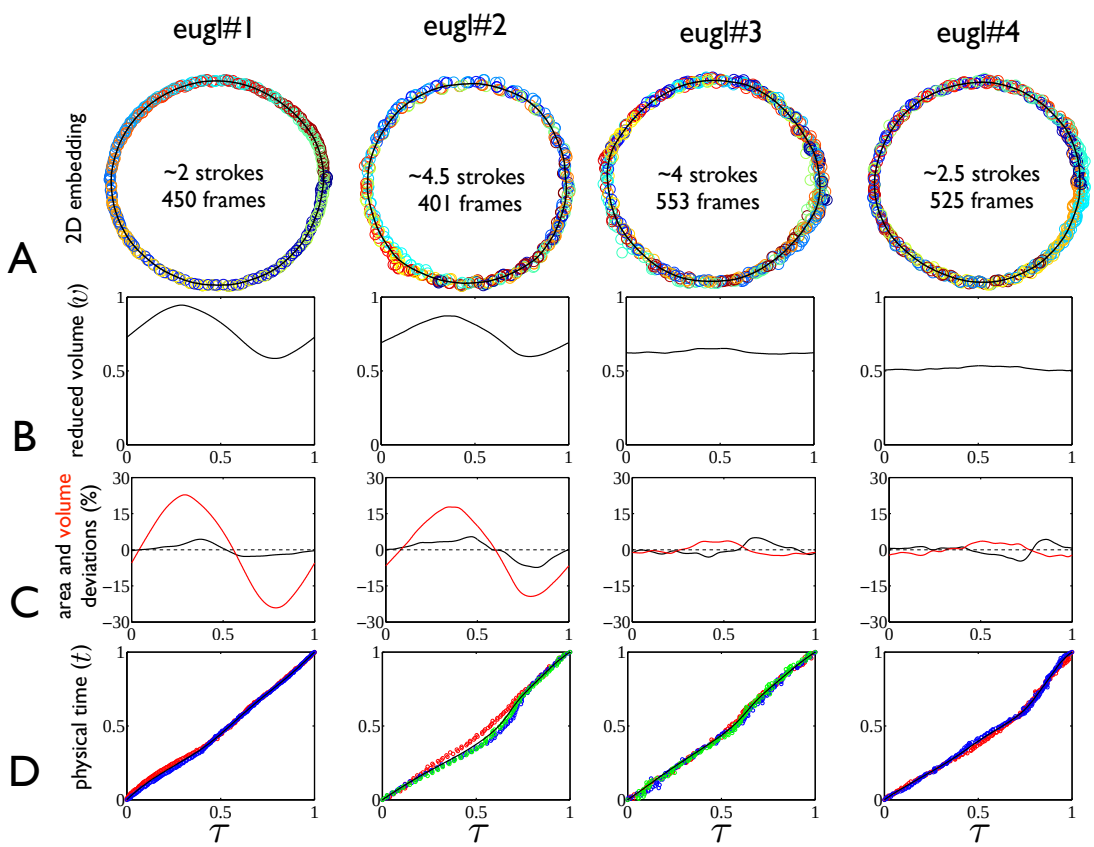


Figure S3: Quantitative analysis of four video recordings: results. (A) 2D embedding of the frames, where the points are colored (from dark blue to red) chronologically from the video recording. The solid black line shows the smooth reconstruction of the stroke path in this embedding. (B) Reduced volume v along the reconstructed stroke. (C) Area and reduced volume deviations relative to the stroke average along the stroke. (D) Relationship between the scaled physical time (t) and arc-length in shape space (pseudo-time τ) during two (euglenids #1 and 4) or three (euglenids #2 and 3) strokes. Data from different strokes are represented in different colors. The spline fit to the data is shown in black.

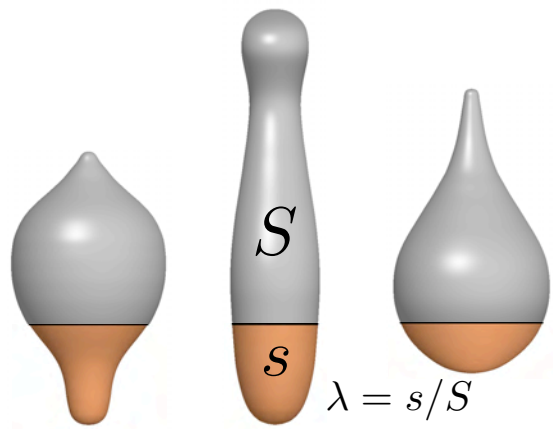


Figure S4: Definition of the body coordinate λ, labeling material parallels. Under the assumption of local area preservation, the total surface area of the pellicle S remains constant during the stroke. At each instant, consider the region delimited by a parallel such that its surface area is a constant s (shown in orange), and define the ratio $\lambda = s/S$ ($= 0.25$ in the figure). By the local area preservation, this parameter labels material parallels along the stroke, shown in black in the figure.

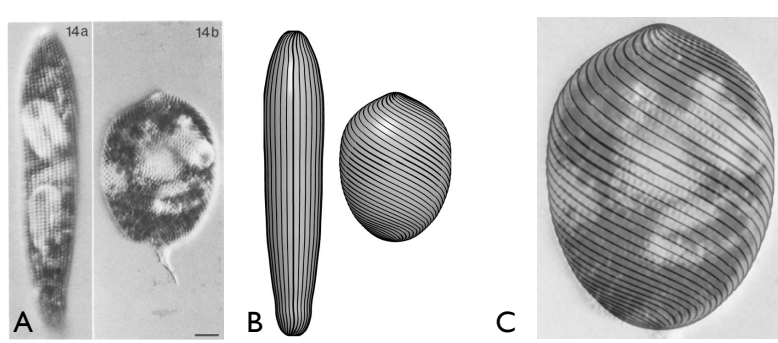


Figure S5: Validation of the theory on the pellicle kinematics with the experiments in [6]. (A) experimental observations, (B) configurations computed with our model, and (C) overlay of computed and observed configurations confirming the accuracy of the model.

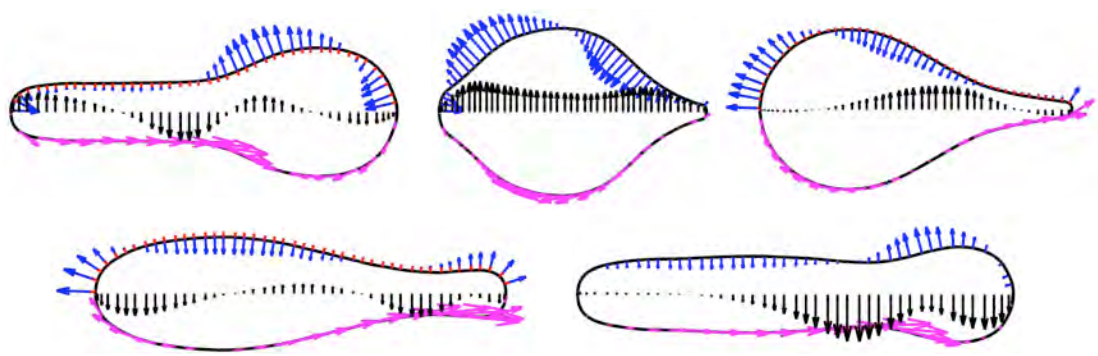
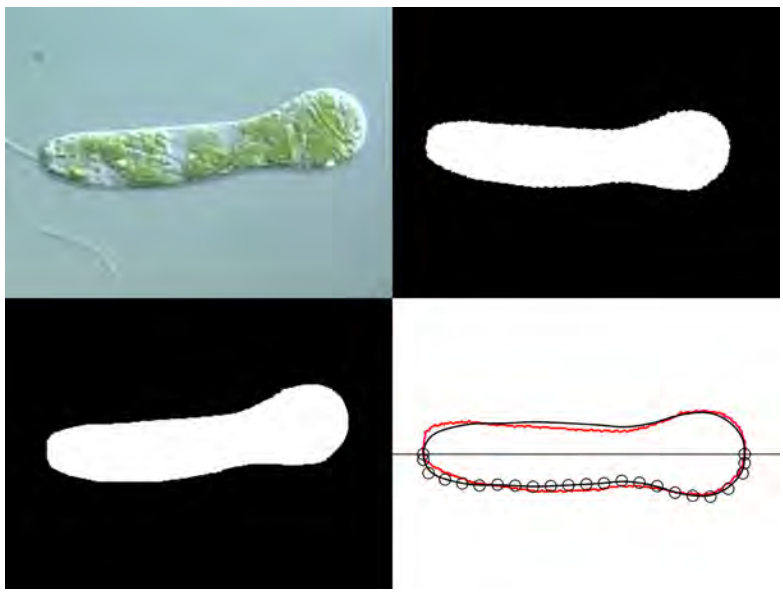
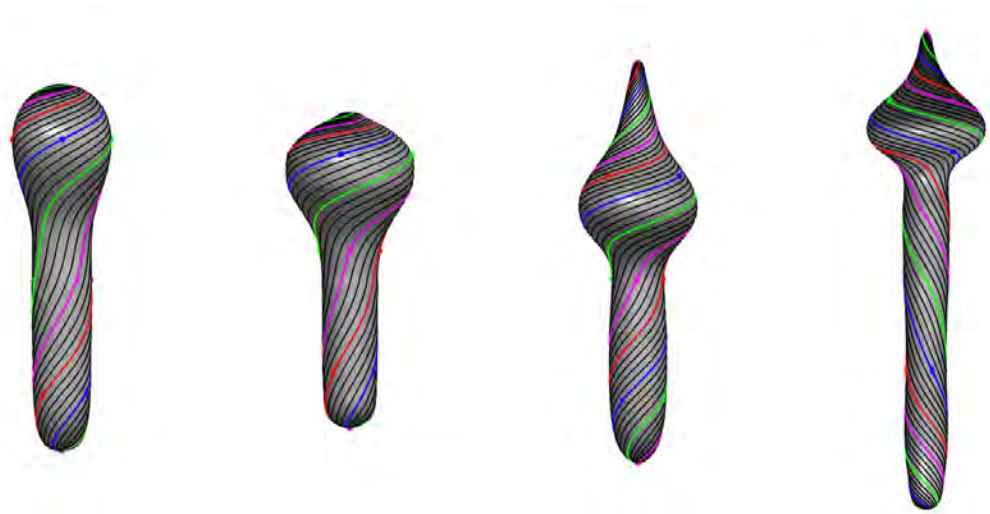


Figure S6: Velocity of the pellicle surface of euglenid #1 along the stroke at five selected instants, computed as described in Section S2.6. All snapshots and velocity components are represented with the same arrow scale. The normal velocities due to shape changes v_n^s and to permeation v_n^p are shown in blue and red, respectively. The tangential velocity along meridians v_λ is shown in magenta, and the azimuthal tangential velocity v_θ is shown in black at a pellicle meridian aligned with the symmetry axis.



Movie S1. Processing of the video recording of euglenid #1. Original frames (top left), segmented frames (bottom left), aligned segmentation (top right), and B-Spline fit to the boundary (bottom right). The red dots represent the boundary pixels, the black curve is the B-Spline fit, and the circles represent the control points of the B-Spline.



Movie S2. The four model euglenids executing metaboly side by side. The net linear and angular motion along the symmetry axis is determined by the hydrodynamical analysis. Material markers, shown on selected pellicle strips shown in color, help appreciate the net angular motion. The four models have been scaled to have the same surface area.

Table S1: (SI) Summary of the kinematic features and swimming performance of the four strokes. U denotes the net displacement in one stroke, in units of body length $2R = \sqrt{S/\pi}$, defined as the diameter of the sphere with the same surface area of the pellicle. Eff_L denotes the Lighthill efficiency, accounting for the outer fluid, or the outer and inner fluids. The power is nondimensionalized by $\bar{P} = 6\pi\eta R^3/T^2$, where T is the duration of one stroke.

Eugl	U	$\Delta\theta/(2\pi)$	$P_{r,z}^{out}/\bar{P}$	$P_{\theta}^{out}/\bar{P}$	Eff_L^{out} (%)	$P_{r,z}^{out+inn}/\bar{P}$	$P_{\theta}^{out+inn}/\bar{P}$	$\mathrm{Eff}_L^{out+inn}$ (%)
# 1	0.16	0.0070	11	3.2	0.71	39	3.6	0.24
# 2	0.27	0.055	15	3.8	1.6	47	4.5	0.57
# 3	0.25	0.086	12	3.5	1.6	46	4.0	0.48
# 4	0.40	0.20	25	6.0	2.0	76	7.3	0.76

Appendix E

Auxiliary methods

1 Affine Isometric Correction

NLDR direct methods based in the minimization procedure of local linear embedding (Roweis and Saul, 2000), as Laplacian eigenmaps (Belkin and Niyogi, 2003), local tangent space alignment (Zhang and Zha, 2005), and Hessian eigenmaps (Donoho and Grimes, 2003), to mention a few, find a $\mathbb{R}^d$ representation of the geometrical structure of the embedded manifold without an isometric relationship with the ambient space in $\mathbb{R}^D$, $d \ll D$. To ameliorate this drawback we have developed a procedure to grant, after an affine transformation, the resulting embedding from a direct method with the best isometric relationship to the original input data. Here we follow the work of Sha and Saul (2005) although we proceed in a different fashion. The method proposed by Sha and Saul (2005) relies given a lower dimensional representation $\Xi = \{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_N\} \subset \mathbb{R}^d$ of a higher dimensional input data set $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N\} \subset \mathbb{R}^D$ an affine transformations which minimizes the stress energy given as

$$E_S(\boldsymbol{T}) = \frac{1}{2}\sum_{a=1}^{N}\sum_{b\in\mathcal{N}^{\boldsymbol{x}_a}}\left(\frac{|\boldsymbol{T}(\boldsymbol{\xi}_a - \boldsymbol{\xi}_b)| - |\boldsymbol{x}_a - \boldsymbol{x}_b|}{|\boldsymbol{x}_a - \boldsymbol{x}_b|}\right)^2,$$

where $\boldsymbol{\xi}_a$ are embedded coordinates obtained from a direct NLDR method, and $\boldsymbol{T}$ is a $d \times d$ matrix with positive determinant. By the polar decomposition theorem, it is sufficient to minimize the function over symmetric positive-definite matrices.

We define the best transformation matrix as the minimizer of the above expression

$$\boldsymbol{T}^* = \arg\min_{T} E_S(\boldsymbol{T}),$$

it means than an additional small problem needs to be solved. The computational cost required to solve this problem is practically unrelated with the number of points or the high dimension where the input data lives.

In order to give explicit expressions for the gradient and Hessian of the stress energy, which are needed to solve the minimization problem, let us first to define an isomorphism $\mathscr{I}$ which transform square matrices to vectors as

$$\begin{aligned}\mathscr{I} : \boldsymbol{T} \in \mathbb{R}^{d\times d} &\longrightarrow \boldsymbol{t} \in \mathbb{R}^{d^2}\\ T_{\alpha\beta} &\longmapsto t_{(\alpha-1)d+\beta}.\end{aligned}$$

Also we define an auxiliary operator $\mathscr{D}$ which take a vector $\boldsymbol{v} \in \mathbb{R}^d$ and return a rectangular matrix in $\mathbb{R}^{d\times d^2}$

$$\begin{aligned}\mathscr{D} : \mathbb{R}^d &\longrightarrow \mathbb{R}^{d\times d^2}\\ \boldsymbol{v} &\longmapsto \sum_{\alpha=1}^{d}\sum_{\beta=1}^{d}(\boldsymbol{e}_\alpha \otimes \boldsymbol{v})(\boldsymbol{e}_\beta \otimes \boldsymbol{E}_{(\alpha-1)d+\beta}),\end{aligned}$$

where $\{\boldsymbol{e}_\alpha\}_{\alpha=1,\ldots,d}$ and $\{\boldsymbol{E}_\alpha\}_{\alpha=1,\ldots,d^2}$ are the canonical basis of $\mathbb{R}^d$ and $\mathbb{R}^{d^2}$, respectively.

Example: given $\boldsymbol{v} \in \mathbb{R}^3$ we can write $\mathscr{D}\boldsymbol{v}$ as a rectangular matrix $\boldsymbol{D}(\boldsymbol{v})$

$$\boldsymbol{D}(\boldsymbol{v}) = \begin{pmatrix} v_1 & v_2 & v_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & v_1 & v_2 & v_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & v_1 & v_2 & v_3 \end{pmatrix},$$

therefore, by using these definitions $\boldsymbol{T}\boldsymbol{\xi} = \mathscr{D}\boldsymbol{\xi} \circ \mathscr{I}\boldsymbol{T} = \boldsymbol{D}(\boldsymbol{\xi})\boldsymbol{t}$, where $\boldsymbol{\xi} \in \mathbb{R}^d$.

Then the stress energy can be rewritten in a more suitable way avoiding to have a matrix as argument, instead we can express the cost function as

$$E_S(\boldsymbol{t}) = \frac{1}{2}\sum_{a=1}^{N}\sum_{b\in\mathcal{N}^a}\left(\frac{|\boldsymbol{\zeta}_{ab}| - |\boldsymbol{x}_{ab}|}{|\boldsymbol{x}_{ab}|}\right)^2,$$

where we have introduced the new variables $\boldsymbol{\zeta}_{ab} = \boldsymbol{\zeta}_a - \boldsymbol{\zeta}_b = \boldsymbol{D}(\boldsymbol{\xi}_a - \boldsymbol{\xi}_b)\boldsymbol{t}$ and $\boldsymbol{x}_{ab} = \boldsymbol{x}_a - \boldsymbol{x}_b$ to make the algebra easier, as it will be clear later on. Furthermore, the transformation matrix have been expressed as a vector by applying the operator $\mathscr{I}$, thus is $\boldsymbol{t} = \mathscr{I}\boldsymbol{T}$.

The gradient and Hessian of the stress energy can be computed explicitly which able us to solve the problem with a Newton-Raphson method. Then the gradient of the stress energy is as follows

$$\frac{\partial E_S(\boldsymbol{t})}{\partial \boldsymbol{t}} = \sum_{a=1}^{N}\sum_{b\in\mathcal{N}^a}\left(\frac{|\boldsymbol{\zeta}_{ab}| - |\boldsymbol{x}_{ab}|}{|\boldsymbol{x}_{ab}|}\right)\frac{\boldsymbol{\eta}_{ab}}{|\boldsymbol{\zeta}_{ab}|},$$

where $\boldsymbol{\eta}_{ab} = [\boldsymbol{D}(\boldsymbol{\xi}_{ab})]^T\boldsymbol{\zeta}_{ab}$, and $\dfrac{\partial|\boldsymbol{\zeta}_{ab}|}{\partial \boldsymbol{t}} = \dfrac{\boldsymbol{\eta}_{ab}}{|\boldsymbol{\zeta}_{ab}|}$, $\boldsymbol{\eta}_{ab} \in \mathbb{R}^{d^2}$.

Since the Hessian is

$$\frac{\partial^2 E_S(\boldsymbol{t})}{\partial \boldsymbol{t}^2} = \sum_{a=1}^{N}\sum_{b\in\mathcal{N}^a}\left[\frac{\boldsymbol{\eta}_{ab}}{|\boldsymbol{\zeta}_{ab}|}\otimes\frac{\boldsymbol{\eta}_{ab}}{|\boldsymbol{\zeta}_{ab}|} + \left(\frac{|\boldsymbol{\zeta}_{ab}| - |\boldsymbol{x}_{ab}|}{|\boldsymbol{x}_{ab}|}\right)[\boldsymbol{D}(\boldsymbol{\xi}_{ab})]^T[\boldsymbol{D}(\boldsymbol{\xi}_{ab})]\right]\frac{1}{|\boldsymbol{\zeta}_{ab}|\ |\boldsymbol{x}_{ab}|},$$

Finally, the Newton-Raphson algorithm is stated as

$$\boldsymbol{t}^{i+1} = \boldsymbol{t}^i - \boldsymbol{J}^{-1}(\boldsymbol{t}^i)\ \boldsymbol{r}(\boldsymbol{t}^i),$$

where $\boldsymbol{J}(\boldsymbol{t}) = HE_S(\boldsymbol{t})$ is the Hessian, and whereas $\boldsymbol{r}(\boldsymbol{t}) = \nabla E_S(\boldsymbol{t})$ is the gradient. As initial condition we chose the identity matrix $\boldsymbol{t}^0 = \mathscr{I}\boldsymbol{I}$.

*R*emark. To solve this problem we have applied first a L-BFGS algorithm (Nocedal and Wright, 1999) with a coarse tolerance, and then a Newton-Raphson' s method with line search until the error reach an acceptable small value.

2 Augmented Lagrangian for thin shells under Kirchhoff-Love assumptions

The optimality conditions can be obtained from differentiating the Lagrangian function

$$\mathcal{L}(\boldsymbol{P}, \lambda) = \Pi(\boldsymbol{P}) - \lambda\, C(\boldsymbol{P}),$$

where C represents a constraint maintained by the Lagrange multiplier or physical reaction λ. After defining a new set of variables $\tilde{\boldsymbol{x}} = (\boldsymbol{P}, \lambda) = (\boldsymbol{P}_1, \boldsymbol{P}_2, ..., \boldsymbol{P}_N, \lambda)$, the optimal solution of this saddle-point problem can be sought with the Newton-Raphson method applied to the nonlinear set of equations given by $\partial_{\boldsymbol{P}}\mathcal{L} = 0$, $\partial_{\lambda}\mathcal{L} = 0$. However, this approach may lead to mere stationary points, not minimizers of the elastic energy, physically unstable equilibria. Furthermore, given the difficulty in setting good initial guesses for the Lagrange multipliers, this solution strategy is not robust.

A robust strategy that guarantees stable equilibria is based on the augmented Lagrangian method, which combines the standard Lagrangian with penalties. This method retains the exactness of the Lagrange multipliers method and the minimization principle of penalty methods. The minimization is performed iteratively on the positions of the control points only for frozen Lagrange multipliers, which are updated explicitly (see Nocedal and Wright (1999), and Conn et al. (1991) for further details). The augmented Lagrangian for thin shells under Kirchhoff-Love assumptions can be stated as

$$\mathcal{L}_A(P, \Lambda) = \Pi(P) - \sum_i \boldsymbol{\lambda}_i^u \cdot \boldsymbol{C}_i^u(P) + \frac{1}{2}\sum_i \frac{|\boldsymbol{C}_i^u(P)|^2}{\mu_i^u} - \sum_i \lambda_i^\theta C_i^\theta(P) + \frac{1}{2}\sum_i \frac{\left[C_i^\theta(P)\right]^2}{\mu_i^\theta},$$

where $\Lambda = \{\boldsymbol{\lambda}_1^u, \ldots, \boldsymbol{\lambda}_K^u, \mu_1^u, \ldots, \mu_K^u, \lambda_1^\theta, \ldots, \lambda_L^\theta, \mu_1^\theta, \ldots, \mu_L^\theta\}$, such that $\boldsymbol{\lambda}_i^u \in \mathbb{R}^3$ account for the Lagrange's multipliers associated to the displacement constraints $\boldsymbol{C}_i^u \in \mathbb{R}^3$ along $\partial\Omega_u$, whereas the $\lambda_i^\theta \in \mathbb{R}$ are related with constraints C_i^θ of the shell slope on the boundary curve $\partial\Omega_\theta$. Finally, the $\mu_i^u \in \mathbb{R}$ and $\mu_i^\theta \in \mathbb{R}$ are the penalty parameters for the displacement and rotation constraints, respectively. Full description of the constraints $\boldsymbol{C}_i^u$ and C_i^θ to apply essential boundary conditions of displacement and rotation, respectively, in the context of Kirchhoff-Love theory can be found in the work by Millán et al. (2012).

We solve all thin shell problems with the augmented Lagrangian method such that a minimizer consistent with the constraints is found. We proceed in two stages, first we find an approximate minimizer through an L-BFGS algorithm with coarse tolerance, and then the approximation is refined by Newton-Raphson's method with a line-search (Brent's algorithm) to find the minimum for a given descent direction.